

# A Prediction Strategy based on Decision Variable Analysis for Dynamic Multi-objective Optimization

Jinhua Zheng<sup>a,b</sup>, Yubing Zhou<sup>a,\*</sup>, Juan Zou<sup>a</sup>, Shengxiang Yang<sup>c</sup>, Junwei Ou<sup>a</sup>, Yaru Hu<sup>a</sup>

<sup>a</sup>*Key Laboratory of Intelligent Computing and Information Processing (Ministry of Education), Xiangtan University, Xiangtan, Hunan, 411105, China*

<sup>b</sup>*Hunan Provincial Key Laboratory of Intelligent Information Processing and Application, Hengyang Normal University, Hengyang, 421002, China*

<sup>c</sup>*School of Computer Science and Informatics, De Montfort University, Leicester LE1 9BH, U.K.*

---

## Abstract

Many multi-objective optimization problems in reality are dynamic, requiring the optimization algorithm to quickly track the moving optima after the environment changes. Therefore, response strategies are often used in dynamic multi-objective algorithms to find Pareto optimal. In this paper, we propose a hybrid prediction strategy based on the classification of decision variables, which consists of three steps. After detecting the environment change, the first step is to analyze the influence of each decision variable on individual convergence and distribution in the new environment. The second step is to adopt different prediction methods for different decision variables. Finally, adaptive selection is applied to the solution set generated in the first and second steps, and solutions with good convergence and diversity are selected to make the initial population more adaptable to the new environment. The prediction strategy can help the solution set converge while maintaining its diversity. The experimental results and performance show that the proposed algorithm is capable of significantly improving the dynamic optimization performance compared with five state-of-the-art evolutionary algorithms.

---

\*Corresponding author

*Email addresses:* jhzheng@xtu.edu.cn (Jinhua Zheng), 872258528@qq.com (Yubing Zhou), zoujuan@xtu.edu.cn (Juan Zou), syang@dmu.ac.uk (Shengxiang Yang), junweiou@163.com (Junwei Ou), huyaru1199@gmale.com (Yaru Hu)

*Keywords:*

Dynamic multi-objective optimization, Evolutionary algorithms, Decision  
Variable Analysis, Adaptive Selection, Diversity

---

## 1. Introduction

Many real-world multi-objective optimization problems(MOPs) are dynamic, meaning their objectives conflict with each other and/or parameter may change over time. Over the past decades, many multi-objective evolutionary algorithms(MOEAs) have achieved success on MOPs. However, when faced with complex dynamic multi-objective optimization problems(DMOPs) [1], traditional MOEAs [2] [3] [4] have obvious deficiencies. In DMOPs, the Pareto optimal front(*POF*) and the Pareto optimal set(*POS*) are not always fixed, and the optimal population that evolves in a traditional MOEA may lose diversity. Especially in the later stages of evolution, the population gradually converges, which means that the population has difficulty adapting to the new environment. In order to effectively track the changing *POF* and/or *POS* over time, MOEAs should be able to respond quickly to environmental changes.

Since MOEAs cannot effectively solve DMOPs, many researchers have done a lot of work on DMOPs, such as detecting whether the environment changes [5] [6] [7] [8]. Some dynamic multi-objective optimization algorithms(DMOEAs) have also been proposed to track a moving *POF* or *POS* quickly and obtain a *POS* that is uniformly distributed over time. On the other hand, in recent years, DMOEAs have been extensively applied in many areas, such as scheduling [9] [10], control [11] [12], planning [13] [14] and design [15], for example, Deb et.al [9] combines NSGA-II with response strategy to effectively solve the hydro-thermal power scheduling. In a hydro-thermal power scheduling, the demand of power is unequal for all concerned units in different time, so the change of the problem is dynamic over time and the solutions of problem must be found when there is a change in power demand. Compared with traditional MOEAs, the algorithm with response strategy can respond to environmental changes more quickly and

help the algorithm jump out of local optimum. Most existing DMOEAs are constructed by combining classic MOEAs with efficient dynamic techniques, including predictive-based and diversity-based approaches. These technologies are competitive in addressing DMOPs, but each technology has usually been limited to solving specific types of problems. For example, prediction-based methods typically achieve better performance on DMOPs with predictable characteristics. Diversity-based approaches have advantages in dealing with DMOPs with dynamically changing characteristics that can lead to a severe loss of diversity. Each method has unique advantages for a specific type of problem.

However, if only the existing diversity generation strategies are adopted, we can not effectively guide the population to explore decision space, and the convergence of the algorithm is greatly reduced. If only the prediction method is used, the performance of the algorithm depends on the training results of the prediction model. In order to avoid these defects as much as possible and combine the advantages of the two methods, we propose a prediction strategy based on decision variable analysis(DVA). When an environmental change is detected, the influence of each decision variable on individual convergence and distribution in the new environment is analyzed, and then combined with historical information, different strategies are adopted to re-initialize the different decision variables. Finally, using adaptive selection to select the initial population with better convergence and distribution solutions for the new environment, we hope that the prediction strategy can ensure that the population can adapt to the environment quickly while maintaining good diversity. Experimental results show that DVA is more adaptable to changing environments than the five algorithms. The main contributions of this study are summarized as follows.

- A decision variable analysis method is used to effectively analyze the impact of decision variables on individuals in the new environment. Combining with historical information, the initial population is generated in different ways according to different decision variables, which can avoid the loss of diversity when the prediction strategy promotes population

convergence.

- We propose an angle-based adaptive selection method to select solutions with better convergence and distribution to form the initial population, so as to better balance the convergence and distribution of the population.
- DVA is a response strategy that can be combined with other classical algorithms to effectively handle DMOPs.

The paper is organized as follows. Section II provides a review of related work on DMOPs and motivation. The proposed prediction strategy based on DVA and the general framework for the proposed method is detailed in Section III. Section IV introduces the test instances, compared algorithms, performance metric and parameter settings. The results and discussions are provided in Section V and Section VI. Finally, conclusions are drawn in Section VII.

## 2. Related Work

In this paper, we consider that minimization problems and DMOPs [1] [16] [17] can be presented as follows:

$$\begin{cases} \min F(x, t) = \{f_1(x, t), f_2(x, t), \dots, f_m(x, t)\}, \\ s.t. g(x, t) \leq 0, h(x, t) = 0, \end{cases} \quad (1)$$

where  $x = (x_1, x_2, \dots, x_n)$  is the  $n$ -dimensional decision variables within the decision space  $\Omega$ ;  $F = (f_1, f_2, \dots, f_m)$  is the  $m$ -dimensional objective vector;  $g(\mathbf{x}, t) \leq 0$  and  $h(x, t) = 0$  are the inequality and equality constraints. The time variable  $t$  [18] is associated with the generation number of the EA, and can be calculated as follows:

$$t = \frac{1}{n_t} \lfloor \frac{\tau}{\tau_t} \rfloor, \quad (2)$$

where  $\tau$  is the generation number;  $n_t$  is change severity, and  $\tau_t$  is change frequency. There are some definitions of DMOPs as follows:

**Definition 1.** *Pareto Dominance [2]:* Assume that  $p$  and  $q$  are any two individuals in the population;  $p$  is said to dominate  $q$ , written as  $f(p) \prec f(q)$ , if  
80  $f_i(p) \leq f_i(q) \forall i \in 1, 2, \dots, m$  and  $f_j(p) < f_j(q) \exists j \in 1, 2, \dots, m$ .

**Definition 2.** *Pareto Optimal Set (POS):*  $x$  is the decision vector;  $\Omega$  is the decision space;  $F$  is the objective function. A solution is said to be nondominated if it is not dominated by any other solutions in  $\Omega$ . Thus, the *POS* is the set of  
85 all nondominated solutions and can be defined mathematically as follows:

$$POS := \{x \in \Omega | \neg \exists x^* \in \Omega, F(x^*) \prec F(x)\}. \quad (3)$$

**Definition 3.** *Pareto Optimal Front (POF):*  $x$  is the decision vector;  $\Omega$  is the decision space;  $F$  is the objective function. Thus, the *POF* is the set of all nondominated solutions with respect to the objective space and can be defined mathematically as follows:

$$POF := \{y = F(x) | x \in POS\}. \quad (4)$$

90 On the other hand, based on the dynamic changes of the *POS* and *POF*, Farina et al. [2] classified DMOPs into four different types.

- *Type I:* The *POS* changes with time but the *POF* is fixed.
- *Type II:* Both the *POS* and *POF* change with time.
- *Type III:* The *POS* remains fixed, while the *POF* changes with time.
- 95 • *Type IV:* Both the *POS* and *POF* remain fixed.

We mainly deal with the first three types of changes in DMOPs, although the *Type IV* change may also occur in some cases.

In recent years, DMOEAs [19] [20] [21] [22] have been proposed to deal with DMOPs. The majority can be classified as diversity-based approaches, convergence-based approaches or other approaches [23] [18] [24]. Convergence-  
100 based approaches focus on enhancing population convergence. Current convergence-based methods include the memory-based strategy [25] [26] [27] [28], the predictive strategy [29] [30] [31] [32] [33] [34]. Memory-based methods respond quickly

to environmental changes by recording past historical information. When *POF*  
105 or *POS* changes periodically, it usually achieves the best results. On the other  
hand, The predictive strategy responds to environmental changes by combining  
other methods, such as strategies in machine learning [35] [22]. However, it  
usually requires a training cycle, and the performance of such models may be  
poor.

110 Unlike convergence-based approaches, diversity-based methods focus on main-  
taining the population diversity. When an environmental change is detected,  
the decision variable space can be better searched. Generally, diversity-based  
approaches can be divided into two categories according to the way diversity is  
enhanced; these are diversity introduction and diversity preservation. Diversity  
115 introduction method [36] [37] [38] is mainly used to respond to environmental  
changes. Specifically, when environmental changes are detected, diversity in-  
troduction method is used to generate some solutions to increase the diversity  
of the population. The latter method [39] [23] pays attention to the diversity  
preservation of the algorithm itself. Such methods rely only on static evolution  
120 capabilities to find a set of optimal solutions, so the convergence ability of the  
algorithm is lacking. Among many kinds of DMOEAs, we mainly consider the  
following two categories:

#### A. Diversity Introduction Methods

The diversity introduction method mainly considers the loss of potential di-  
125 versity in a dynamic environment, especially at a later stage of an environmental  
moment, when the population almost converges. If an environmental change is  
detected, the population loses its ability to explore the entire decision space due  
to convergence to the *POS*, so the population cannot quickly converge to the  
new *POS*. In order to solve this problem, many DMOEAs have been proposed in  
130 recent years. For example, Deb et al. extended NSGA-II [2] into two algorithm-  
s (DNSGA-II-A and DNSGA-II-B) [9]. When a change in the environment is  
detected, DNSGA-II-A will randomly re-initialize 20% of the individuals, and  
another algorithm DNSGA-II-B will randomly mutate 20% of the individuals,  
both of which can increase the diversity of the population to some extent. Harri-

son et al. [40] proposed dynamic vector evaluated particle swarm optimization. Based on this, Helbig et al. [41] proposed a heterogeneous dynamic vector evaluated particle swarm optimisation(HDVEPSO) algorithm. When the objective function changes, HDVEPSO randomly re-initializes 30 percent of the particles to avoid falling into local optimum. Goh et al. [39] introduced a competitive-cooperative coevolutionary algorithm (dCOEA) where random individuals are generated to enhance diversity of the population if the environment changes. In conclusion, these diversity introduction methods randomly generate or mutate individuals to prevent the population from falling into local optimum after detecting changes in the environment. However, most of these methods increase population diversity by randomly reinitializing or mutating only some of the individuals, and cannot effectively explore the new decision space. Additionally, the historical information cannot be used effectively, so the algorithm does not converge fast.

#### B.Prediction-based Methods

The prediction strategy uses historical information to generate an initial population to help the population adapt quickly to the environment. For different DMOPs, a suitable predictive model is critical. A good predictive model can provide a guiding direction for the evolution of the population towards the *POF*. Hatzakis and Wallace [42] proposed a feed-forward prediction strategy (FPS) that predicts only the boundary points of the population, which makes it difficult to deal with DMOPs with nonlinear correlations between decision variables. Zhou and Jin et al. [43] proposed a population prediction strategy(PPS). Both FPS and PPS use the autoregressive model to predict population. In addition, due to the lack of historical information in the early stage, PPS convergence is very slow. Ruan et al. [44] applied gradual search to predict the idea position of the individual in the new environment. Jiang et al. [45] suggested a steady-state and generational evolutionary algorithm (SGEA), which guides the search of solutions by a moving direction from the centroid of the non-dominated solution set to the centroid of the whole population. Recently, Cao et al [34] presents a novel prediction model combined with a multiobjective

evolutionary algorithm based on decomposition to solve dynamic multiobjective optimization problems. In short, these prediction strategies can enhance the convergence speed of the population in the new environment to a certain extent. However, there may be problems in predicting so errors may occur occasionally and some prediction strategies require additional training processes,  
170 which consumes a lot of computing resources.

From this discussion, we can see that the two methods have their own advantages in resolving different aspects of DMOPs. On the premise of introducing the method of DVA, in this study, we hybridize the diversity introduction methods and the prediction-based methods to improve the quality of solutions. The  
175 proposed algorithm is described next.

### 3. Proposed Algorithm

In this section, we propose a new algorithm for solving DMOPs, which aims to generate a population close to the true *POF* while maintaining the ability  
180 of the population to explore the decision space. The main idea is to analyze the impact of each decision variables on individuals at the current environment and to adopt different strategies to optimize them. To achieve this target, we use a simple decision variable analysis method when the environment changes, and then adopt different generation methods for different decision variables to  
185 generate the initial population.

#### 3.1. Decision variable analysis method

The DVA method analyzes the influence of each decision variables on individuals and then performs simple classification processing when the change is detected. The purpose is to find distribution-related decision variables, and  
190 other decision variables are considered to be related to convergence. Therefore, we randomly generate  $n$  representative solutions representing  $n$ -dimensional decision variables, and then disturb the decision variables represented by each representative solution. The values of other decision variables remain unchanged.



Each representative solution generates  $nPer$  perturbation solutions through perturbation. Finally, the representative solution and its perturbation solution are ranked by nondominated sort [2]. We judge the influence of the dimension decision variables on individuals in the current environment by the dominance relationship between the representative solution and the perturbation solution.

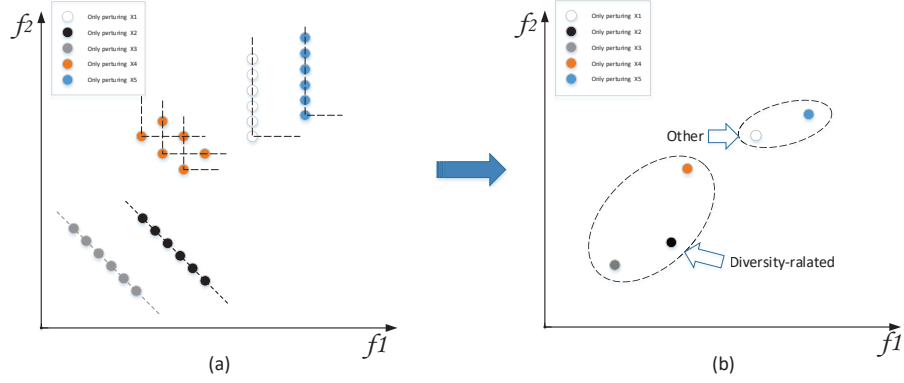


Figure 1: Simple analysis of the impact of each dimension of decision variables

Fig.1 presents an example to illustrate the main idea of the DVA, where a bi-objective minimization problem with five decision variables  $x_1, x_2, x_3, x_4$  and  $x_5$  is considered. A number of  $n$  representative solutions (five in this example) are first randomly generated. Then, a number of  $nPer$  (six in this example) perturbations are performed on each of the five variables of the representative solutions. Fig.1.(a) illustrates the objective value of the representative solutions and the solutions generated by the perturbations. A nondominated sort is performed for each representative solution and the perturbation solution it generates. As shown in Fig1(a), there are three situations:

- The relationship between the representative solution and the disturbance solution is nondominated (the gray and black points in Fig 1.(a)), so the decision variables representing the dimension can only affect the distribution of individuals in the current environment, but cannot affect the convergence of individuals.

- The relationship between the representative solution and the perturbation solution is dominated (the blue and white points in Fig 1.(a)), so the dimensional decision variable can only affect the individual convergence under the current environment, but cannot affect the individual distribution.
- Both dominated and nondominated relations exist between the representative solution and the perturbation solution (the orange points in Fig 1.(a)). The representative decision variable can affect both the distribution and the convergence of the individual.

For these three cases, the decision variables in the first and second cases can affect the individual distribution, so the dimensional decision variables are classified as distributed correlation (DV), and the other decision variables are classified as the set of other decision variables (OV, which can only affect the individual convergence), as shown in figure 1.(b). The specific algorithm flow is described in detail in algorithm 1.

### 3.2. The Prediction Strategy Based On DVA

The prediction strategy in this section is to find the solution set as close as possible to the true *POS* when a change is detected. Different from other prediction strategies, in this paper, different prediction methods are used according to different decision variables to generate a set with sufficient capacity to explore the decision space and move closer to the true *POS*. The proposed algorithm determines the strategy adopted by the per-dimensional decision variables in the current environment based on the information obtained from the DVA of the previous moment and the current time environment. If the  $i_{th}$  decision variable belongs to  $OV_t$  at the previous moment and the current moment belongs to  $DV_{t+1}$ , or the previous moment belongs to  $DV_t$  and the current moment belongs to  $OV_{t+1}$ , then the historical information on the  $i_{th}$  is useless, and the strategy of random initialization is adopted on this dimension. In other cases, different prediction mechanisms are adopted according to the classification of

---

**Algorithm 1** Decision Variable Analysis Method

---

**Input:**

$n$ (the number of decision variable at  $t + 1$  environment time)

**Output:**

$DV$ (Diversity-related),  $OV$ ,  $C$ (a set of perturbations)

```

1: for  $i = 1$  to  $n$  do
2:    $p$ (the representative solution) randomly generate, representing the  $i$ -
     dimensional decision variable.
3:    $C_{tmp}$ (Temporary sets for storing representative solutions and perturba-
     tion solutions)  $\leftarrow \{p\} \cup$  Generated  $nPer$  solutions by random perturbation
     of  $P$  in the  $i_{th}$  dimension
4:   Sort  $C_{tmp}$  using nondominated sort
5:   if  $C_{tmp}$  contains non-dominated relationships then
6:      $DV \leftarrow DV \cup \{i\};$ 
7:   else
8:      $OV \leftarrow OV \cup \{i\};$ 
9:   end if
10:   $C \leftarrow C \cup C_{tmp}$  and empty  $C_{tmp}$ 
11: end for

```

---

the current time. The details of the prediction method are given as follows. For the decision variables belonging to  $DV_{t+1}$ , a hybrid strategy combining Latin hypercube sampling [46] and random sampling is adopted to enhance the exploration ability of the population in the new environment. In Fig 2, an example is presented to illustrate the idea of the hybrid strategy.

This strategy mainly uses the minimum point( $min_t$ ) and the maximum point( $max_t$ ) of the population. Defined as follows:

$$\begin{cases} min_t = (min_t^1, min_t^2, \dots, min_t^n) \\ max_t = (max_t^1, max_t^2, \dots, max_t^n). \end{cases} \quad (5)$$

where  $min_t^i$  and  $max_t^i$  are the minimum and maximum values of  $i_{th}$  decision

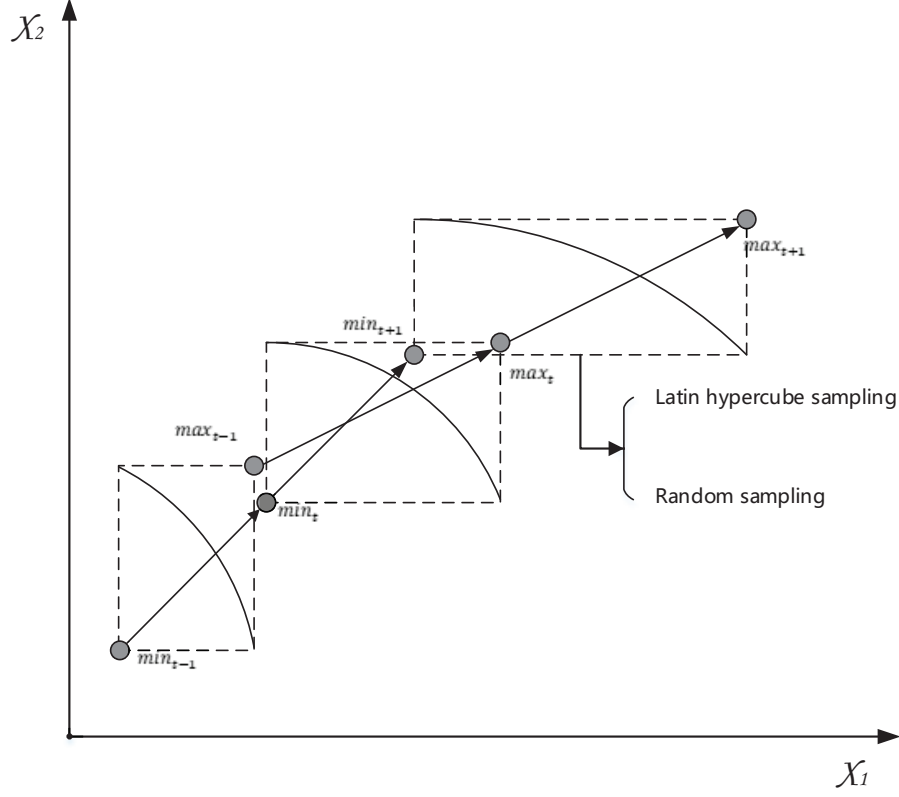


Figure 2: the prediction strategy for DV.

250 variable at time  $t$ . Then the moving direction of the minimum and maximum points in  $i_{th}$  dimension is defined as

$$\begin{cases} \min D_t^i = \min_t^i - \min_{t-1}^i \\ \max D_t^i = \max_t^i - \max_{t-1}^i. \end{cases} \quad (6)$$

Then, the minimum and maximum points at time  $t+1$  can be calculated from the minimum and maximum points at time  $t$  and the direction of movement.

$$\begin{cases} \min_{t+1}^i = \min_t^i + \min D_t^i \\ \max_{t+1}^i = \max_t^i + \max D_t^i. \end{cases} \quad (7)$$

In the space composed of the minimum point and the maximum point at  
255 time  $t+1$ , the Latin hypercube sampling method [46] is adopted to generate

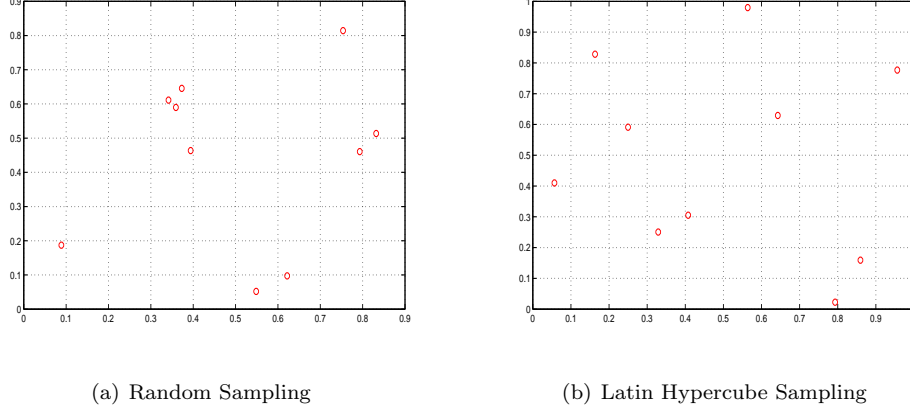


Figure 3: The results of selecting 10 points from 0 to 1 by random sampling and Latin hypercube sampling.

$N_2$  solutions, and then  $N_3$  solutions are randomly generated in the space. The process of Latin hypercube sampling and random sampling is described in Fig 3. Different from random sampling, the Latin hypercube sampling guarantees that there are sample points on every aliquot of each dimension. The combination  
 260 of the two methods can enhance the exploring ability of the initial population in the predicted decision space. If the  $i$ -th decision variable belongs to  $OV_{t+1}$  in the current environment, that is, the dimension decision variable can only affect the convergence of the individual at the current time, then in this dimension we use the central point prediction method to accelerate the convergence speed  
 265 of the population in the new environment. Suppose  $C_t$  is the center point of  $POS_t$ , and  $POS_t$  is the  $POS$  of the population that is finally optimized at time  $t$ , then  $C_t$  can be calculated by the following formula:

$$C_t = \frac{1}{|POS_t|} \sum_{x \in POS_t} x_t. \quad (8)$$

$|POS_t|$  is the number of solutions in  $POS$  at time  $t$ ;  $x_t = x_t^1, x_t^2, \dots, x_t^n$  is the solution at time  $t$ . Therefore, the moving direction  $D_t$  of the center point at

time  $t$  is defined as follows:

$$D_t = C_t - C_{t-1}. \quad (9)$$

Then the value at time  $t + 1$  on the dimension can be generated from the center point and the prediction direction at time  $t$  according to the following formula.

$$x_{t+1} = x_t + D_t. \quad (10)$$

$x_t$  is the final solution obtained before the environmental change, and  $x_{t+1}$  is the predicted solution. The flow of predicting the generation of solution sets is described in algorithm 2 in detail. First,  $N_1$  solutions are generated by the Latin hypercube sampling method combined with the centre point prediction method(Algorithm 2 Lines 1-14), and then  $N_2$  solutions are generated by the random sampling method combined with the centre point prediction method(Algorithm 2 Lines 15-28). The predicted population consists of both solutions.

### 3.3. Adaptive Selection Based On The Angle

The population has good diversity in the decision space does not mean it also has good diversity in the objective space, for some problems (e.g., the deflection problem), the set with good distribution in the decision space will lead to the deterioration of the population in the objective space. Therefore, when the environment changes, in order to allow the initialized population to adapt to the new environment quickly, and to promote population convergence in the objective space while strengthening its distribution, we propose an angle-based adaptive selection method. After the analysis of decision variables and the prediction based on the analysis, the populations  $C$  and  $Q$  generated by algorithms 1 and 2 are merged into population  $G$ . Then the  $\cos(\theta)$  between each individual and other individuals in population  $G$  and the crowding of each individual are calculated(Algorithm 3 Lines 1-3). We use nondominated sort to accelerate the convergence of the population and select individuals from the

---

**Algorithm 2** The Prediction Strategy Based On DVA

---

**Input:**

$DV_t, DV_{t+1}, OV_t, OV_{t+1}$

**Output:**

$Q$ ( a set of individuals generated by prediction )

```
for i = 1 to  $N_2$  do
2:   for j = 1 to  $n$  do
       if  $((j \in DV_t \& j \in OV_{t+1}) || (j \in OV_t \& j \in DV_{t+1}))$  then
4:      $x_i^j$  randomly generated
       else
6:       if  $(j \in DV_{t+1})$  then
            $x_i^j$  Generated by Latin hypercube sampling
8:       else
            $x_i^j$  Calculated by formula 9
10:      end if
       end if
12:   end for
        $Q \leftarrow Q \cup \{i\}$ 
14: end for

for i = 1 to  $N_3$  do
16:   for j = 1 to  $n$  do
       if  $((j \in DV_t \& j \in OV_{t+1}) || (j \in OV_t \& j \in DV_{t+1}))$  then
18:      $x_i^j$  randomly generated
       else
20:       if  $(j \in DV_{t+1})$  then
            $x_i^j$  randomly generated
22:       else
            $x_i^j$  Calculated by formula 9
24:      end if
       end if
26:   end for
        $Q \leftarrow Q \cup \{i\}$ 
28: end for
```

---

295 first layer to join the initial population in turn. At the same time, we enhance  
the distribution of the population through the angle between individuals. If the  
angle between individuals and the selected individuals is less than the threshold  
 $\sigma$ , then these individuals will be ignored in the selection according to the layers.  
If all the layers are selected and the number of initial population is still less  
300 than  $N$ , the individuals with small crowding will be selected to join the initial  
population. Algorithm 3 describes the process of selecting the solutions.

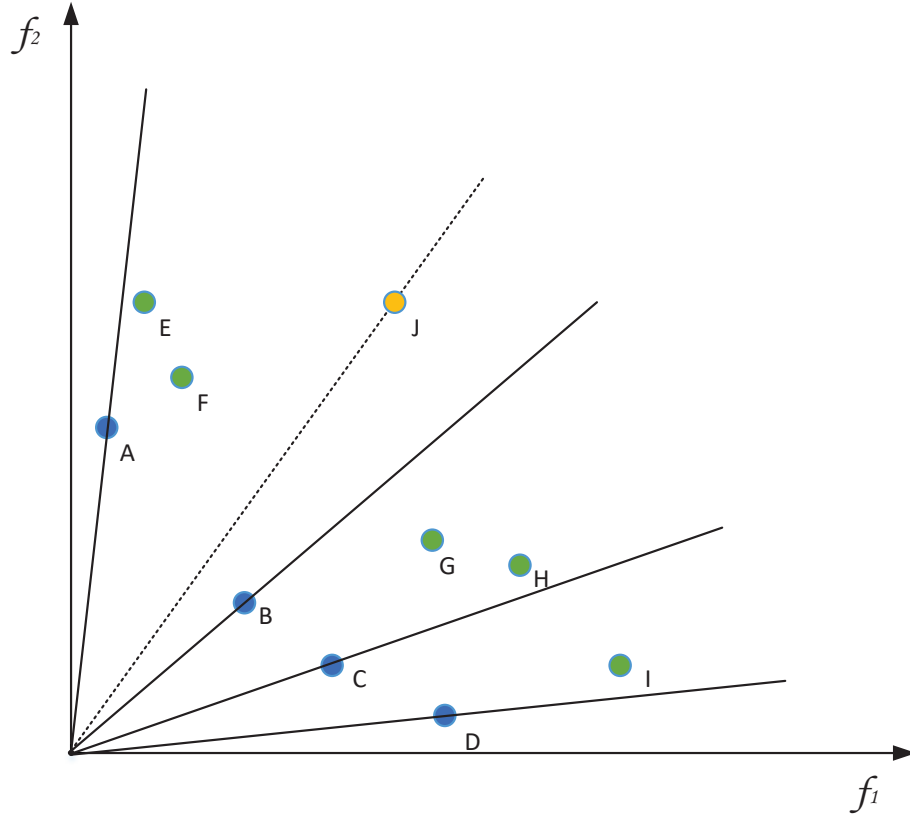


Figure 4: Selection operation.

As shown in fig.4, assuming a population size of 5, first, the  $\cos(\theta)$  between  
each individual and the others is calculated according to formula 11, and the sum  
of the first  $k$  maximum values of  $\cos(\theta)$  is taken as the individual's crowding. As  
305 a simple analysis, if there are more individuals in a certain direction, the greater



---

**Algorithm 3** Adaptive Selection Based On Angle

---

**Input:**

$C$  (Solution Set Generated by Algorithm 1),  $Q$  (Solution Set Generated by Algorithm 2)

**Output:**

$P$ (the initial population at environment time  $t+1$ )

$G \leftarrow C \cup Q$

Calculate the  $\cos(\theta)$  between each individual and others in the population  $G$  according to Equation .

- 3: Calculate the Crowding: the sum of the top  $k$  largest  $\cos$  values of each individual in  $G$  as the individual's crowding

Sort  $G$  using nondominated sort and Get the number of layers  $K$

**for**  $i = 1$  to  $K$  **do**

- 6: Select individual  $x \in G_i$  and  $P \leftarrow P \cup \{x\}$

**for**  $j = 1$  to  $|G|$  **do**

**if**  $(\cos(x, x_j) > \delta)$  **then**

- 9:  $G \leftarrow G \setminus \{x_j\}$

$P_{tmp} \leftarrow P_{tmp} \cup x_j$

**end if**

- 12: **end for**

**end for**

**while**  $|P| < N$  **do**

- 15: Choose the individual  $x$  with the minimum crowding from  $P_{tmp}$ ;

$P \leftarrow P \cup \{x\}$

$P_{tmp} \leftarrow P_{tmp} \setminus \{x\}$

- 18: **end while**
-

the first  $k$   $\cos$  values between individuals in this region and other individuals, that is, the greater the crowding of individuals in that direction; then sort all the individuals using nondominated sort. There are three layers in Fig.4. The first layer has blue points; the second layer has green points, and the last layer has orange points. From the first layer of individuals, individual A is selected to join the population. Since the  $\cos(\theta)$  between the individuals E, F and A is less than the threshold  $\delta$ , E and F are ignored. Similarly, individual B is selected and individual G is ignored. Individual C is selected and individual H is ignored. Individual D is selected and individual I is ignored. The individuals are added to the population according to the number of layers in turn. When all the layers have been selected, if the number of individuals in the population is still less than  $N$ , the ignored individuals will be selected, and the individual J is finally selected.

$$\cos(\theta) = \frac{\sum_{m=1}^M (f_m(x_i) - z_m^*)(f_m(x_j) - z_m^*)}{\sqrt{\sum_{m=1}^M (f_m(x_i) - z_m^*)^2} \sqrt{\sum_{m=1}^M (f_m(x_j) - z_m^*)^2}}. \quad (11)$$

where  $z_m^* = (z_1^*, z_2^*, \dots, z_M^*)$  is the ideal point, and  $z_m^* = \min_{x_i \in Q} f_m(x_i)$ ,  $x_i$  and  $x_j$  are two individuals in the population.

### 3.4. A General Framework

The procedure of the whole algorithm is illustrated in Algorithm 4 in detail. When the environment changes and the response strategy is not used, the impact of the decision variables on the individual is analyzed in the new environment, and then the different strategies in Algorithm 2 are used to target different decision variables based on the analysis results. Finally, the solution sets generated by Algorithms 1 and 2 are adaptively selected to obtain the initial population in the new environment. For optimizing stationary MOP, we adopt the dynamic evolutionary environment model(DEE) [20] because it can better guarantee the convergence and distribution of the algorithm when the target

dimension is not large. Certainly, as a response strategy, it can combine with most classical algorithms to deal with DMOPs and it is mentioned later in the discussion section.

---

**Algorithm 4** Framework of DVA

---

**Input:**

$N$ (population size)

**Output:**

$P$ (population)

Initial population  $P = x_1, x_2, \dots, x_N$

**while** stopping criterion not met **do**

**if** change detected and not responded **then**

4:      $(DV_{t+1}, OV_{t+1}, C) = \text{DVA}(n)(\text{Algorithm 1})$

$Q = \text{the Prediction Strategy Based On DVA}(\text{Algorithm 2})$

$P = \text{ASA}(C, Q)(\text{Algorithm 3})$

**end if**

8:     Optimize population with DEE

**end while**

---

### 3.5. Computational complexity of the compared algorithms and DVA

335     The optimization algorithm consumes the most computational resources of the compared algorithms and DVA. The computational complexity of each optimization algorithm and DVA are analyzed as follows:

(1) DNSGA-II: From the original paper of DNSGA-II [9], the optimization algorithm is NSGA-II [2] and the computational resource is spent on nondominated sorting  $O(M(2N)^2)$ , crowding-distance assignment  $O(M(2N)\log(2N))$  and sorting  $O(2N\log(2N))$ . The overall computational complexity is  $O(MN^2)$ ; 340  
 $M$  is the number of objectives and  $N$  is the population size.

(2) PPS: PPS [43] chooses RM-MEDA [47] as the MOEA optimizer. In RM-MEDA, the computational complexity of RM-MEDA includes modeling, 345  
reproduction and the selection operator. The modeling cost is  $O(nN)$  where

$n$  is the number of the decision space. The reproduction spends  $O(nK)$  and  $K$  is the number of clusters. The selection operation is the same as in NSGA-II [2]. Therefore, the overall computational complexity is  $O(MN^2)$ .

- (3) DMS: The prediction in DMS requires  $O(N)$  computational complexity, and the gradual search strategy and the random diversity maintenance strategy all take  $O(N)$  computational complexity. Then computational complexity of the overall framework of DMS is mainly spent on the nondominated sort. Therefore, the overall computational complexity of DMS for one response mechanism is  $O(MN^2)$ .
- (4) SGEA: SGEA [45] is introduced in section 4.2 and it consumes during steady-state evolution and environmental selection. The whole steady-state evolution part takes  $O(MN^2)$  computations and the environmental selection procedure spends  $O(MN^2)$  computations. Therefore, the overall computational complexity is  $O(MN^2)$ .
- (5) DVA: For the overall framework of each generation, the main computational resource in DVA is consumed by adaptive selection and the prediction strategy. The DVA method also needs computational resources when an environmental change is detected. It requires a total of  $O(nM)$  computations, and the prediction strategy requires  $O(nN)$  computations. The adaptive selection (line 6 of Algorithm 4) requests  $O(M(N_1 + N)^2)$ , where  $M$  is the number of objectives,  $N_1$  is the population size of  $C$  in algorithm 3 and  $N$  is the population size. Therefore, the overall computational complexity of DVA in each generation is  $O(M(N_1 + N)^2)$ .

## 4. Experimental Design

### 4.1. Test instances

In this section, 16 dynamic frequently used multi-objective test problems are adopted to examine the performance of the algorithm. The test problems include the FDA test suite(FDA1-FDA4), DMOP test suite(DMOP1-DMOP3) and JY test suite(JY1-JY9). The FDA test suite, which was proposed by Farina et al [1],

375 is the basic test problem in DMOPs. The DMOP test suite was extended from  
the FDA suite by Goh et al. [39]. The two test suites have linearly correlated  
decision variables and both are the most commonly used test suite in dynamic  
multi-objectives and are widely used to test the performance of DMOEAs [43]  
[44] [45]. The JY test suite is the new DMOPs test suite proposed by Jiang et  
380 al [17]. More and more researchers pay attention to JY test suite and apply it  
to test the performance of DMOEA [48] [49]. It not only has linear correlation  
between decision variables, but also has nonlinear correlation on some issues.  
The JY test suite also introduces some complex features that are useful for  
examining the performance of the algorithm.

#### 385 4.2. Compared Algorithms

We compared the performance of the proposed prediction strategy with five  
popular DMOEAs: the random Initialization Strategy(RIS), the dynamic non-  
dominated sorting genetic algorithm II (DNSGA-II) [9], the population predic-  
tion strategy (PPS) [43], the effect of diversity maintenance on prediction(DMS)  
390 [44] and a steady-state and generational evolutionary algorithm (SGEA) [45].

(1) DNSGA-II: NSGA-II [2] is a classic algorithm for multi-objective opti-  
mization. To apply it to DMOPs, Deb et al. modified the NSGA-II algorithm  
to adapt to different environments. When it is detected that the environment  
changes, some individuals are randomly initialized, and some individuals are  
395 replaced with mutated solutions to adapt to the new environment.

(2) PPS: PPS divides the population into two parts: the center point and  
manifold. When the environment changes, the univariate autoregression(AR)  
model is used to predict the position of the next population center point. Sim-  
ilarly, the next manifold is generated from the previous manifold. The initial  
400 population at the next environment time is obtained from the predicted center  
point plus the manifold.

(2) DMS: DMS consists of three parts, exploration based on prediction, a  
gradual search strategy and a random diversity maintenance strategy. The first  
part is mainly to accelerate the convergence of the population, while the second

405 and third parts focus on maintaining the distribution of the population.

(4) SGEA: SGEA can take advantage of the fast and steady tracking ability of steady-state algorithms and the good diversity preservation of generational algorithms for solving DMOPs. Some information from the previous environment and new environment are used for reacting to environmental changes when  
410 a change is detected.

### 4.3. Performance Metrics

In this section, performance metrics, which can evaluate convergence, distribution and diversity of the obtained solution set, are introduced.

1)Generational Distance (GD): Van Veldhuizen [39] [27] presented the GD  
415 metric, which measures the population's convergence. The GD indicator is defined as follows:

$$GD(POF_t, P_t) = \frac{\sum_{v \in P_t} d(POF_t, v)}{|P_t|}, \quad (12)$$

where  $d(POF_t, v) = \min_{u \in POF_t} \sqrt{\sum_{j=1}^m (f_j^v - f_j^u)^2}$  is the minimum Euclidian distance between  $v$  and the point in  $POF_t$ .  $POF_t$  is a set of uniformly distributed Pareto optimal points in the  $POF$  at time  $t$ ;  $P_t$  is the solution obtained by  
420 the algorithms.

2)Inverted Generational Distance(IGD): IGD [43] [44] is a metric which assesses the convergence and diversity of the obtained solution set. The IGD is calculated as follows:

$$IGD(POF_t, P_t) = \frac{\sum_{v \in POF_t} d(v, P_t)}{|POF_t|}, \quad (13)$$

where  $d(v, P_t) = \min_{u \in P_t} \sqrt{\sum_{j=1}^m (f_j^v - f_j^u)^2}$  is the minimum Euclidian distance  
425 between  $v$  and the point in  $P_t$ .  $POF_t$  is a set of uniformly distributed Pareto optimal points in the  $POF$  at time  $t$ ;  $P_t$  is the solution obtained by the algorithms. The IGD [43] performance metric is a comprehensive index and is developed to measure the convergence and diversity of the algorithm' obtained solutions.

430 3) Schott's spacing metric (SP): This kind of indicator was developed by Schott [50] to investigate the distribution of the found Pareto front. The SP is calculated as follows:

$$SP = \sqrt{\frac{1}{|P_t| - 1} \sum_{i=1}^{|P_t|} (D_i - \bar{D})^2}, \quad (14)$$

where  $D_i$  is the Euclidean distance between the  $i$ th member in  $P_t$ , and its nearest member in  $P_t$ , and  $\bar{D}$  is the average value of  $D_i$ . SP measures how evenly the  
435 solutions in  $|P_t|$  are distributed.

#### 4.4. Parameter Settings

In order to fairly compare algorithms, most parameters of the compared algorithms were set to the values used in their original papers. The experimental parameters were set as follows. The severity of change was fixed to 10 and  
440 the frequency of change was set to 20, 25 and 30; The population size was  $N = 100$  and the dimensions of the test problem's decision space were  $n = 10$ ; In general, 5% of the population was randomly selected and re-evaluated to detect environmental changes. Each algorithm ran independently 20 times on all problems, and there were 50 environmental changes. The crossover probability  
445 was  $p_c = 0.8$  and the mutation probability was  $p_m = 1/n$  for all algorithms.

- Parameters in DVA:  $N_2 = 0.8N$  and  $N_3 = 0.2N$  in the prediction strategy and  $\delta = 0.9998$  in the adaptive selection strategy.

## 5. Experimental Results and Analysis

In order to analyze the ability of the algorithm to solve DMOPs, the average IGD, GD and SP results over a series of time windows were obtained and  
450 their standard deviation values are presented in Tables1 ,2 and.3, respectively, where the best values obtained by one of five algorithms are highlighted in bold face. Additionally, the Wilcoxon rank-sum test [51] was carried out to indicate significance between different results at the 0.05 significance level.

455 5.1. Results on FDA and dMOP problems

It can be obtained from Table.1 that DVA achieved the best results on most issues of FDA and dMOP. The IGD metric mainly depends on the closeness, distribution, and coverage of an approximation to the true *POF*. Smaller IGD values represent better performance of the algorithm. We can use IGD together  
460 with SP and GD to deeply and extensively reveal the algorithm's performance on the test instances. The IGD of DVA was the best in most of the test problems except for FDA4 and DMOP1. This indicates that DVA had a better distribution and convergence than the other methods. For FDA4, the IGD of DVA was the best when  $\tau_t=20$  and 25, but when  $\tau_t$  increased to 30, SGEA was  
465 better than DVA. It shows that the distribution and convergence of DVA were weaker than SGEA in dynamic changes. As for DMOP1, the IGD of DVA was second only to SGEA, and with the increase of  $t$ , DVA was better than SGEA. The conclusion can be drawn that DVA performs moderately on problems like FDA4.

470 As shown in Table.2, DVA had the smallest values of GD on the FDA1, FDA4, dMOP2 and dMOP3. The smaller values of GD imply that the algorithm had better convergence than the other algorithms. For most problems, DVA significantly performed better than RIS, DNSGA-II, PPS and DMS. However, when compared with SGEA, DVA performed worse on FDA2, FDA3 and d-  
475 MOP1. The reason is that the *POS* of FDA2 and dMOP1 remain fixed. SGEA preserves half of the solutions from the last population, so the values of GD on those test problems are the smallest. As for FDA3, the density of solutions on the *POF* can be varied.

For the SP metric, as can be seen from Table.3, DVA was better than the  
480 other five algorithms on FDA1, DMOP2 and DMOP3. However, on FDA2 and dMOP1, the performance of DMS and PPS was significantly better than SGEA and DVA. This is because the *POS* of FDA2 and dMOP1 are fixed, and DMS and PPS have more advantages in dealing with these problems. On the three-dimensional problem of FDA4, DMS, DVA and SGEA all show better  
485 distribution. As for FDA3, the distribution of DMS is obviously better than



the other algorithms because the gradual search strategy and random diversity maintenance strategy of the DMS can help it maintain better diversity on this problem.

In addition to the indicators represented in the table.1, we provide an evolution curve of the average IGD value of the test examples in Fig.5. As can be seen, compared with the other algorithms, DVA can respond to changes more quickly, and had better stability in most test problems, especially in FDA4, so it was able to obtain higher convergence performance. However, on dMOP2 and dMOP3, the stability of DVA was not as good as that of SGEA. The reason may be that SGEA is very capable of tracking environmental changes, whereas DVA is weaker. However, compared with the other algorithms, DVA still had better stability and convergence. Overall, DVA achieved good performance in the FDA and dMOP test suites.

## 5.2. Results on JY problems

Unlike the FDA and dMOP test suites, JY problems are a new benchmark suite with several complex characteristics including a nonmonotonic and time-varying relationship among decision variables. In addition, the types of some problems change over time in the optimization process. The following observations are drawn from Table.1. DVA significantly performed best over other approaches on most JY problems except for JY6 and JY7 in terms of the IGD metric. JY6 is a multimodal problem, where not only the number of local optima changes over time, but also the *POS* is dynamically shifted. Similarly, JY7 is a multimodal problem, and different from JY6, the number of local optima remains fixed, and the overall *POF* shape can be concave or convex. These tests mainly tests the search ability of the algorithm in the whole decision space. Combined with SP in Table.3, compared with other algorithms, it is clear that DVA has good distribution on JY6 and JY7. At the same time, from the convergence metric GD in Table.2, we can see that the convergence of DVA on these two issues is far less than that of SGEA. This also results in DVA being inferior to SGEA in the comprehensive metric IGD. The reason is that DVA cannot

Table 1: Mean and SD of IGD indicator obtained by six algorithms.

Prob	$(\tau_t, n_t)$	RIS	DNSGA-II [9]	PPS [43]	DMS [44]	SGEA [45]	DVA
FDA1	(20,10)	8.32e-2(6.21e-3)‡	7.74e-3(2.35e-4)‡	1.53e-2(9.30e-3)‡	8.62e-3(5.74e-4)‡	6.62e-3(3.33e-4)‡	<b>4.87e-3(5.78e-4)</b>
	(25,10)	6.30e-2(6.52e-3)‡	6.65e-3(8.09e-5)‡	1.87e-2(1.14e-2)‡	8.18e-3(2.15e-3)‡	5.82e-3(1.98e-4)‡	<b>4.73e-3(3.52e-4)</b>
	(30,10)	4.16e-2(6.95e-3)‡	6.17e-3(9.76e-5)‡	1.53e-2(6.26e-3)‡	6.55e-3(5.31e-4)‡	5.25e-3(1.46e-4)‡	<b>4.74e-3(3.84e-4)</b>
FDA2	(20,10)	4.35e-2(1.18e-3)‡	3.65e-1(2.98e-4)‡	3.21e-2(6.44e-4)‡	3.18e-2(2.31e-4)‡	2.86e-2(3.27e-5)‡	<b>2.84e-2(1.03e-4)</b>
	(25,10)	3.77e-2(9.70e-4)‡	3.64e-1(1.14e-4)‡	3.14e-2(1.18e-4)‡	3.19e-2(5.06e-4)‡	2.86e-2(5.66e-5)‡	<b>2.84e-2(8.95e-5)</b>
	(30,10)	3.55e-2(5.70e-4)‡	3.64e-1(3.78e-5)‡	3.12e-2(1.52e-4)‡	3.15e-2(6.17e-4)‡	2.86e-2(1.75e-5)‡	<b>2.84e-2(5.98e-5)</b>
FDA3	(20,10)	4.76e-2(3.44e-3)‡	4.59e-2(1.81e-3)‡	2.00e-2(1.11e-2)‡	3.11e-2(7.18e-3)‡	1.37e-2(2.02e-3)‡	<b>9.08e-3(5.52e-4)</b>
	(25,10)	9.87e-2(1.66e-3)‡	9.33e-2(1.08e-3)‡	8.33e-2(5.05e-4)‡	8.17e-2(2.08e-3)‡	6.44e-2(8.54e-4)‡	<b>6.26e-2(3.15e-4)</b>
	(30,10)	2.00e-2(1.30e-3)‡	4.50e-2(4.13e-4)‡	8.77e-3(1.44e-3)‡	1.65e-2(3.18e-3)‡	9.56e-3(1.51e-3)‡	<b>8.26e-3(1.26e-3)</b>
FDA4	(20,10)	1.17e-1(2.67e-3)‡	1.01e-1(1.51e-3)‡	8.60e-2(1.60e-3)‡	8.44e-2(1.82e-3)‡	7.00e-2(4.82e-4)‡	<b>6.32e-2(5.88e-4)</b>
	(25,10)	2.70e-2(1.44e-3)‡	4.52e-2(7.16e-4)‡	1.41e-2(5.90e-3)‡	2.49e-2(9.04e-3)‡	1.16e-2(1.99e-3)‡	<b>8.67e-3(8.50e-4)</b>
	(30,10)	8.76e-2(1.81e-3)‡	8.65e-2(1.29e-3)‡	8.19e-2(1.36e-3)‡	7.86e-2(8.26e-4)‡	<b>6.17e-2(4.95e-4)</b>	6.24e-2(4.28e-4)
dMOP1	(20,10)	5.63e-1(4.57e-2)‡	3.24e-2(1.12e-2)‡	9.95e-3(4.06e-3)‡	<b>8.70e-3(2.09e-3)</b>	1.00e-2(3.33e-3)	1.41e-2(5.91e-3)
	(25,10)	3.94e-1(3.05e-2)‡	2.36e-2(1.23e-2)‡	1.19e-2(1.09e-2)‡	<b>7.70e-3(3.42e-3)</b>	9.13e-3(3.10e-3)	1.04e-2(7.20e-3)
	(30,10)	3.51e-1(1.31e-2)‡	1.44e-2(7.91e-4)‡	6.86e-3(2.20e-3)‡	6.77e-3(2.05e-3)‡	9.58e-3(2.42e-3)‡	<b>6.49e-3(1.80e-3)</b>
dMOP2	(20,10)	1.76e-1(2.02e-2)‡	5.18e-2(9.15e-3)‡	4.99e-2(6.72e-2)‡	1.27e-2(2.72e-3)‡	1.18e-1(1.52e-4)‡	<b>5.17e-3(7.49e-4)</b>
	(25,10)	1.24e-1(1.81e-2)‡	3.70e-2(1.31e-2)‡	1.80e-2(1.91e-2)‡	1.18e-2(2.51e-3)‡	1.17e-1(1.16e-4)‡	<b>5.17e-3(5.79e-4)</b>
	(30,10)	1.05e-1(1.75e-2)‡	3.40e-2(1.84e-2)‡	5.49e-2(1.04e-1)‡	8.99e-3(1.25e-3)‡	1.47e-1(2.85e-4)‡	<b>4.88e-3(9.45e-4)</b>
dMOP3	(20,10)	7.92e-2(1.26e-2)‡	7.78e-3(2.04e-4)‡	2.62e-2(1.95e-2)‡	8.34e-3(7.87e-4)‡	6.62e-3(3.33e-4)‡	<b>4.67e-3(2.46e-4)</b>
	(25,10)	6.15e-2(5.34e-3)‡	6.69e-3(1.08e-4)‡	1.86e-2(8.16e-3)‡	7.37e-3(5.73e-4)‡	5.82e-3(1.98e-4)‡	<b>4.79e-3(2.19e-4)</b>
	(30,10)	4.36e-2(8.61e-3)‡	6.06e-3(3.91e-5)‡	1.07e-2(9.08e-3)‡	6.31e-3(5.97e-4)‡	5.25e-3(1.46e-4)‡	<b>4.79e-3(4.21e-4)</b>
JY1	(20,10)	7.01e-2(5.68e-3)‡	6.32e-1(5.36e-2)‡	4.59e-2(1.31e-2)‡	1.45e-2(4.90e-3)‡	1.22e-2(8.35e-4)‡	<b>5.99e-3(5.36e-4)</b>
	(25,10)	4.38e-2(3.09e-3)‡	5.49e-1(2.53e-1)‡	1.85e-2(2.20e-2)‡	1.54e-2(3.83e-3)‡	1.01e-2(7.75e-4)‡	<b>6.06e-3(3.91e-4)</b>
	(30,10)	2.87e-2(1.20e-3)‡	2.82e-1(5.48e-2)‡	1.30e-2(1.30e-2)‡	1.09e-2(1.66e-3)‡	8.49e-3(5.80e-4)‡	<b>5.94e-3(4.33e-4)</b>
JY2	(20,10)	8.60e-2(4.48e-3)‡	5.93e-2(8.70e-4)‡	9.02e-2(3.66e-2)‡	5.13e-2(2.87e-4)‡	5.18e-2(6.56e-4)‡	<b>5.03e-2(2.33e-4)</b>
	(25,10)	6.54e-2(2.37e-3)‡	5.58e-2(5.67e-4)‡	6.15e-2(1.99e-2)‡	5.19e-2(1.83e-3)‡	5.10e-2(6.87e-4)‡	<b>5.03e-2(3.78e-4)</b>
	(30,10)	5.84e-2(1.16e-3)‡	5.43e-2(3.62e-4)‡	5.75e-2(1.22e-2)‡	<b>5.03e-2(5.42e-4)</b>	5.07e-2(6.07e-4)‡	5.04e-2(2.76e-4)
JY3	(20,10)	3.35e-1(1.95e-3)‡	<b>2.83e-1(3.37e-3)</b>	3.16e-1(8.01e-4)‡	3.15e-1(7.06e-4)‡	3.39e-1(1.18e-2)‡	3.14e-1(2.95e-3)
	(25,10)	3.25e-1(2.61e-3)‡	<b>2.79e-1(2.30e-3)</b>	3.15e-1(9.03e-4)‡	3.15e-1(8.23e-4)‡	3.39e-1(6.27e-3)‡	3.12e-1(3.07e-3)
	(30,10)	3.20e-1(2.16e-3)‡	<b>2.80e-1(3.00e-3)</b>	3.15e-1(9.57e-4)‡	3.14e-1(3.60e-4)‡	3.36e-1(3.34e-3)‡	3.12e-1(1.76e-3)
JY4	(20,10)	6.94e-2(2.92e-3)‡	3.97e-1(2.49e-2)‡	8.08e-2(3.15e-2)‡	3.13e-2(5.89e-3)‡	2.25e-2(5.27e-4)‡	<b>2.11e-2(8.95e-4)</b>
	(25,10)	4.54e-2(1.73e-3)‡	3.44e-1(5.97e-3)‡	5.22e-2(3.00e-2)‡	3.16e-2(4.50e-3)‡	2.14e-2(2.80e-4)‡	<b>2.06e-2(3.98e-4)</b>
	(30,10)	3.28e-2(1.46e-3)‡	2.94e-1(1.24e-2)‡	2.67e-2(3.81e-3)‡	2.44e-2(1.21e-3)‡	2.09e-2(2.13e-4)‡	<b>2.08e-2(3.61e-4)</b>
JY5	(20,10)	4.73e-2(2.50e-3)‡	3.48e-2(9.64e-4)‡	6.62e-3(1.46e-3)‡	5.10e-3(5.39e-4)‡	4.32e-3(7.60e-5)‡	<b>4.21e-3(2.63e-5)</b>
	(25,10)	2.83e-2(2.38e-3)‡	3.31e-2(4.65e-4)‡	5.36e-3(7.72e-4)‡	4.69e-3(4.50e-5)‡	4.22e-3(3.36e-5)‡	<b>4.21e-3(1.55e-5)</b>
	(30,10)	1.92e-2(2.17e-3)‡	3.31e-2(9.65e-4)‡	5.80e-3(1.58e-3)‡	4.64e-3(4.61e-5)‡	<b>4.18e-3(2.03e-5)</b>	4.22e-3(5.34e-5)
JY6	(20,10)	1.09e+0(4.73e-2)‡	3.43e+0(2.83e-1)‡	2.02e+0(2.35e-1)‡	1.53e+0(1.11e-1)‡	<b>1.84e-1(8.11e-3)</b>	3.98e-1(2.84e-2)
	(25,10)	6.93e-1(1.66e-2)‡	3.03e+0(1.88e-1)‡	1.95e+0(1.11e-1)‡	1.52e+0(1.50e-1)‡	<b>1.28e-1(2.76e-3)</b>	3.64e-1(5.69e-2)
	(30,10)	4.74e-1(9.42e-3)‡	2.63e+0(1.12e-1)‡	1.80e+0(1.51e-1)‡	1.37e+0(1.33e-1)‡	<b>9.28e-2(6.08e-3)</b>	3.25e-1(5.31e-2)
JY7	(20,10)	1.01e+0(4.42e-2)‡	4.42e+0(5.47e-1)‡	5.59e+0(3.31e+0)‡	3.07e+0(5.15e-1)‡	<b>4.67e-1(1.22e-1)</b>	9.04e-1(3.96e-1)
	(25,10)	5.50e-1(2.03e-2)‡	3.17e+0(7.77e-1)‡	4.83e+0(2.87e+0)‡	2.81e+0(5.74e-1)‡	<b>3.86e-1(1.48e-1)</b>	7.59e-1(3.59e-1)
	(30,10)	3.46e-1(4.59e-2)‡	2.59e+0(9.62e-1)‡	2.16e+0(2.61e+0)‡	2.61e+0(5.53e-1)‡	<b>3.85e-1(1.41e-1)</b>	5.30e-1(4.06e-1)
JY8	(20,10)	3.12e-2(8.52e-4)‡	1.15e-1(8.24e-2)‡	8.49e-3(2.89e-4)‡	8.55e-3(7.50e-4)‡	1.80e-2(2.26e-3)‡	<b>8.01e-3(2.50e-4)</b>
	(25,10)	2.15e-2(6.69e-4)‡	1.53e-1(7.92e-2)‡	8.57e-3(3.09e-4)‡	8.39e-3(2.11e-4)‡	1.77e-2(3.03e-3)‡	<b>7.75e-3(3.05e-4)</b>
	(30,10)	1.63e-2(8.33e-4)‡	1.28e-1(7.41e-2)‡	8.50e-3(7.29e-4)‡	8.31e-3(2.73e-4)‡	1.61e-2(1.44e-3)‡	<b>7.52e-3(2.84e-4)</b>
JY9	(20,10)	2.47e-1(2.24e-2)‡	1.48e+0(1.57e-1)‡	2.92e-2(8.31e-3)‡	9.46e-2(3.34e-2)‡	4.65e-2(4.91e-3)‡	<b>1.25e-2(1.65e-3)</b>
	(25,10)	1.16e-1(8.26e-3)‡	9.91e-1(2.89e-1)‡	1.66e-2(9.01e-4)‡	7.51e-2(1.36e-2)‡	3.75e-2(9.42e-4)‡	<b>1.33e-2(3.15e-3)</b>
	(30,10)	6.95e-2(3.58e-3)‡	8.08e-1(1.41e-1)‡	<b>1.15e-2(3.55e-3)</b>	4.36e-2(7.95e-3)‡	3.50e-2(7.64e-4)‡	1.36e-2(1.28e-3)

‡ and † indicate DVA performs significantly better than and equivalently to the corresponding algorithm, respectively.

take into account the convergence well while guaranteeing the distribution in the multimode problem.

From Table.2, it can be seen that DVA had better convergence performance on most JY problems. However, on JY4, the convergence performance of DVA was inferior to SGEA. As for JY6 and JY7, the performance of DVA was worse than SGEA. The reason is shown in the above paragraph. From the distribution indicator of Table 3, we see that the distributions of DVA in JY5 and JY8 test were not as good as those of SGEA and DMS. The reason is similar to that of FDA2 and dMOP1, the *POS* of JY5 and JY8 are fixed. As mentioned, DVA cannot treat this kind of problem as well as SGEA and DMS, but DVA was still

better than the rest of the algorithms on this problem.

Similarly, in addition to the indicators shown in the table.1, we provide the evolution curve of the average IGD value of the JY test problem in Figure.6. We can see that, DVA was able to respond to environmental changes fast and stably in most cases. On JY3, JY4 and JY5, SGEA has roughly similar curves, and both DVA and SGEA were superior to the other algorithms. However, on JY6 and JY7, DVA was significantly inferior to SGEA, DVA's performance was similar to NSGA-II and significantly better than RIS, DMS and PPS. The specific reasons are as analyzed above. As for other issues on JY test suites, DVA did greater advantages.

Table 2: Mean and SD of GD indicator obtained by six algorithms.

Prob	$(\tau_t, n_t)$	RIS	DNSGA-II [9]	PPS [43]	DMS [44]	SGEA [45]	DVA
FDA1	(20,10)	5.69e-2(3.38e-3)†	5.97e-3(3.32e-4)‡	1.91e-2(1.61e-2)‡	7.96e-3(8.85e-4)‡	4.58e-3(1.76e-4)‡	<b>2.57e-3(6.56e-4)</b>
	(25,10)	2.47e-2(1.85e-3)‡	4.59e-3(1.86e-4)‡	2.75e-2(2.27e-2)‡	7.47e-3(2.93e-3)‡	3.87e-3(2.82e-5)‡	<b>2.41e-3(3.63e-4)</b>
	(30,10)	1.32e-2(7.13e-4)‡	3.76e-3(5.93e-5)‡	2.17e-2(1.38e-2)‡	5.21e-3(7.07e-4)‡	3.22e-3(5.90e-5)‡	<b>2.41e-3(4.29e-4)</b>
FDA2	(20,10)	6.93e-2(1.56e-3)‡	3.26e-1(2.67e-3)‡	6.30e-2(7.14e-4)‡	6.01e-2(3.14e-4)‡	<b>4.38e-2(3.50e-4)</b>	5.54e-2(1.31e-3)
	(25,10)	6.10e-2(1.43e-3)‡	3.26e-1(8.80e-4)‡	6.34e-2(5.84e-4)‡	6.18e-2(4.96e-4)‡	<b>4.37e-2(2.68e-4)</b>	5.63e-2(9.35e-4)
	(30,10)	6.03e-2(9.91e-4)‡	3.26e-1(1.77e-3)‡	6.41e-2(2.37e-4)‡	6.34e-2(8.62e-4)‡	<b>4.40e-2(1.52e-4)</b>	5.65e-2(3.10e-4)
FDA3	(20,10)	7.90e-2(3.62e-3)‡	8.74e-2(2.01e-3)‡	6.55e-2(2.05e-2)‡	6.05e-2(8.34e-3)‡	<b>3.92e-2(1.22e-3)</b>	8.42e-2(1.52e-3)
	(25,10)	4.74e-2(8.90e-4)‡	9.34e-2(1.95e-3)‡	6.15e-2(1.51e-2)‡	5.22e-2(1.29e-2)‡	<b>4.30e-2(8.59e-4)</b>	8.50e-2(1.56e-3)
	(30,10)	3.59e-2(1.92e-3)‡	9.58e-2(1.86e-3)‡	7.58e-2(9.58e-3)‡	<b>4.08e-2(4.30e-3)</b>	4.25e-2(1.29e-3)‡	8.39e-2(2.06e-3)
FDA4	(20,10)	8.09e-2(2.62e-3)‡	1.01e-1(2.85e-3)‡	7.02e-2(3.58e-3)‡	6.62e-2(2.90e-3)‡	3.83e-2(5.53e-4)‡	<b>9.23e-3(3.21e-4)</b>
	(25,10)	5.63e-2(2.05e-3)‡	7.47e-2(3.74e-3)‡	6.61e-2(1.31e-3)‡	5.93e-2(5.74e-3)‡	2.99e-2(5.89e-4)‡	<b>8.68e-3(2.74e-4)</b>
	(30,10)	4.14e-2(1.82e-3)‡	5.97e-2(3.48e-3)‡	6.35e-2(3.74e-3)‡	5.24e-2(2.95e-3)‡	2.50e-2(3.07e-4)‡	<b>8.54e-3(2.34e-4)</b>
dMOP1	(20,10)	4.76e-1(6.74e-2)‡	2.65e-2(5.39e-3)‡	8.02e-3(2.52e-3)‡	9.70e-3(4.90e-3)‡	<b>1.45e-3(3.67e-4)</b>	5.75e-3(2.52e-3)
	(25,10)	1.63e-1(5.13e-3)‡	1.53e-2(1.18e-3)‡	9.02e-3(1.01e-2)‡	4.15e-3(7.68e-4)‡	<b>1.23e-3(1.43e-4)</b>	4.39e-3(3.16e-3)
	(30,10)	7.38e-2(3.11e-3)‡	1.23e-2(3.32e-4)‡	2.70e-3(1.58e-4)‡	2.77e-3(3.78e-4)‡	<b>1.07e-3(3.57e-5)</b>	4.02e-3(2.31e-3)
dMOP2	(20,10)	8.95e-2(7.43e-3)‡	4.04e-2(5.12e-3)‡	5.48e-2(7.10e-2)‡	1.37e-2(4.39e-3)‡	1.19e-1(3.07e-4)‡	<b>2.70e-3(8.51e-4)</b>
	(25,10)	3.49e-2(4.31e-3)‡	2.41e-2(2.11e-3)‡	1.44e-2(1.35e-2)‡	1.25e-2(3.46e-3)‡	1.18e-1(3.30e-4)‡	<b>2.71e-3(6.76e-4)</b>
	(30,10)	2.16e-2(2.16e-3)‡	1.62e-2(1.20e-3)‡	4.70e-2(8.46e-2)‡	7.66e-3(1.69e-3)‡	3.36e-3(8.01e-5)‡	<b>2.34e-3(3.69e-4)</b>
dMOP3	(20,10)	5.56e-2(2.19e-3)‡	5.80e-3(1.67e-4)‡	4.33e-2(4.27e-2)‡	8.24e-3(8.28e-4)‡	4.58e-3(1.76e-4)‡	<b>2.34e-3(2.59e-4)</b>
	(25,10)	2.43e-2(2.12e-3)‡	4.53e-3(1.40e-4)‡	2.61e-2(1.69e-2)‡	6.31e-3(9.34e-4)‡	3.87e-3(5.82e-5)‡	<b>2.48e-3(2.18e-4)</b>
	(30,10)	1.39e-2(1.06e-3)‡	3.72e-3(2.64e-4)‡	1.13e-2(1.37e-2)‡	4.95e-3(8.13e-4)‡	3.22e-3(5.90e-5)‡	<b>2.46e-3(4.67e-4)</b>
JY1	(20,10)	7.37e-2(6.66e-3)‡	2.92e+1(8.40e-1)‡	7.75e-2(2.43e-2)‡	1.78e-2(9.20e-3)‡	3.80e-3(2.55e-4)‡	<b>2.39e-3(5.49e-4)</b>
	(25,10)	2.83e-2(3.80e-3)‡	2.74e+1(4.28e+0)‡	2.44e-2(3.81e-2)‡	1.85e-2(4.19e-3)‡	2.99e-3(1.58e-4)‡	<b>2.39e-3(3.47e-4)</b>
	(30,10)	1.22e-2(1.15e-3)‡	2.64e+1(2.55e+0)‡	1.64e-2(2.62e-2)‡	9.74e-3(2.24e-3)‡	2.51e-3(6.70e-5)‡	<b>2.27e-3(4.48e-4)</b>
JY2	(20,10)	8.76e-2(7.85e-3)‡	4.99e-1(9.13e-2)‡	1.21e-1(6.11e-2)‡	5.39e-2(1.04e-3)‡	5.13e-2(3.12e-4)‡	<b>5.01e-2(3.16e-4)</b>
	(25,10)	5.60e-2(3.87e-3)‡	3.40e-1(8.22e-2)‡	7.08e-2(3.74e-2)‡	5.44e-2(3.23e-3)‡	5.10e-2(3.47e-4)‡	<b>5.02e-2(2.35e-4)</b>
	(30,10)	5.06e-2(6.24e-4)‡	4.81e-1(4.69e-2)‡	6.25e-2(2.07e-2)‡	5.13e-2(6.30e-4)‡	5.10e-2(2.87e-4)‡	<b>5.04e-2(4.59e-4)</b>
JY3	(20,10)	2.92e-1(3.47e-3)‡	5.41e-1(2.73e-1)‡	2.26e-1(4.05e-3)‡	2.32e-1(3.31e-3)‡	1.82e-1(1.14e-2)‡	<b>1.59e-1(7.03e-2)</b>
	(25,10)	2.61e-1(4.05e-3)‡	7.39e-1(1.46e-1)‡	2.30e-1(2.59e-3)‡	2.34e-1(1.95e-3)‡	1.78e-1(9.34e-3)‡	<b>9.09e-2(6.01e-2)</b>
	(30,10)	2.46e-1(6.04e-3)‡	5.03e-1(2.65e-1)‡	2.33e-1(2.05e-3)‡	2.35e-1(2.08e-3)‡	1.83e-1(7.74e-3)‡	<b>1.06e-1(7.49e-2)</b>
JY4	(20,10)	6.02e-2(2.39e-3)‡	1.82e+1(5.90e-1)‡	1.02e-1(5.05e-2)‡	2.48e-2(6.07e-3)‡	<b>4.45e-3(8.98e-5)</b>	5.17e-3(1.04e-3)
	(25,10)	3.09e-2(2.10e-3)‡	1.91e+1(6.46e-1)‡	5.73e-2(4.52e-2)‡	2.22e-2(5.63e-3)‡	<b>3.24e-3(1.76e-4)</b>	4.52e-3(6.73e-4)
	(30,10)	1.42e-2(1.39e-3)‡	1.94e+1(5.61e-1)‡	1.85e-2(5.38e-3)‡	1.28e-2(1.62e-3)‡	<b>2.58e-3(5.80e-5)</b>	4.32e-3(4.11e-4)
JY5	(20,10)	4.46e-2(9.40e-4)‡	1.69e+0(2.71e-1)‡	5.29e-3(2.11e-3)‡	2.62e-3(4.16e-4)‡	1.19e-3(1.13e-4)‡	<b>9.17e-4(2.45e-5)</b>
	(25,10)	2.07e-2(1.39e-3)‡	1.81e+0(3.33e-1)‡	3.23e-3(1.33e-3)‡	2.23e-3(1.52e-4)‡	1.08e-3(7.79e-5)‡	<b>9.00e-4(2.07e-5)</b>
	(30,10)	1.09e-2(6.92e-4)‡	2.40e+0(3.27e-1)‡	3.86e-3(2.29e-3)‡	2.02e-3(1.22e-4)‡	1.06e-3(8.95e-5)‡	<b>9.17e-4(5.97e-5)</b>
JY6	(20,10)	3.04e+0(7.71e-2)‡	6.76e+0(3.64e-1)‡	5.56e+0(4.90e-1)‡	4.43e+0(3.81e-1)‡	<b>2.10e-1(1.33e-2)</b>	5.93e-1(4.36e-2)
	(25,10)	2.06e+0(5.54e-2)‡	7.52e+0(1.62e+0)‡	4.44e+1(1.35e+1)‡	1.26e+1(2.57e+0)‡	<b>4.82e-1(1.85e-1)</b>	1.71e+0(8.26e-1)
	(30,10)	1.09e+0(4.25e-2)‡	4.79e+0(1.88e-1)‡	4.87e+0(2.42e-1)‡	3.63e+0(1.81e-1)‡	<b>7.58e-2(1.06e-2)</b>	4.80e-1(8.76e-2)
JY7	(20,10)	3.86e+0(1.48e-1)‡	1.03e+1(1.03e+0)‡	2.97e+1(1.53e+1)‡	1.62e+1(3.24e+0)‡	<b>6.12e-1(2.32e-1)</b>	2.02e+0(8.96e-1)
	(25,10)	1.43e-1(2.07e-2)‡	1.66e-1(8.85e-2)‡	4.75e-3(3.18e-4)‡	4.47e-3(1.67e-4)‡	<b>2.93e-3(1.80e-4)</b>	3.91e-3(1.10e-4)
	(30,10)	1.00e+0(4.14e-2)‡	5.89e+0(1.88e+0)‡	1.07e+1(1.34e+1)‡	1.14e+1(1.87e+0)‡	<b>5.58e-1(2.44e-1)</b>	1.23e+0(9.32e-1)
JY8	(20,10)	2.78e-2(1.22e-3)‡	1.40e-1(8.91e-2)‡	4.95e-3(1.75e-4)‡	4.55e-3(2.08e-4)‡	<b>2.94e-3(1.09e-4)</b>	4.04e-3(2.11e-4)
	(25,10)	1.43e-2(1.11e-3)‡	1.66e-1(8.85e-2)‡	4.75e-3(3.18e-4)‡	4.47e-3(1.67e-4)‡	<b>2.93e-3(1.80e-4)</b>	3.91e-3(1.10e-4)
	(30,10)	8.38e-3(4.11e-4)‡	1.24e-1(6.76e-2)‡	4.97e-3(1.09e-3)‡	4.50e-3(7.14e-5)‡	<b>3.07e-3(8.84e-5)</b>	3.99e-3(1.26e-4)
JY9	(20,10)	4.23e-1(3.46e-2)‡	1.73e+2(4.32e+0)‡	5.59e-2(2.49e-2)‡	1.82e-1(5.56e-2)‡	3.55e-2(4.95e-3)‡	<b>7.58e-3(1.63e-3)</b>
	(25,10)	1.29e-1(2.07e-2)‡	1.67e+2(8.88e+0)‡	2.47e-2(4.20e-3)‡	1.48e-1(2.62e-2)‡	2.88e-2(9.78e-4)‡	<b>1.01e-2(4.66e-3)</b>
	(30,10)	5.07e-2(6.06e-3)‡	1.62e+2(1.49e+1)‡	1.39e-2(8.10e-3)‡	8.23e-2(1.67e-2)‡	2.70e-2(1.68e-4)‡	<b>9.35e-3(1.14e-3)</b>

‡ and † indicate DVA performs significantly better than and equivalently to the corresponding algorithm, respectively.

Table 3: Mean and SD of SP indicator obtained by six algorithms.

Prob	$(\tau_t, n_t)$	RIS	DNSGA-II [9]	PPS [43]	DMS [44]	SGEA [45]	DVA
FDA1	(20,10)	1.02e-2(6.36e-4)†	5.56e-3(4.09e-4)†	5.27e-3(1.10e-3)†	4.51e-3(6.13e-4)†	4.21e-3(1.27e-4)†	<b>3.67e-3(7.45e-5)</b>
	(25,10)	7.67e-3(2.48e-4)†	6.17e-3(1.31e-3)†	5.38e-3(1.19e-3)†	4.05e-3(5.05e-4)†	3.94e-3(7.22e-5)†	<b>3.61e-3(8.07e-5)</b>
	(30,10)	6.67e-3(2.88e-4)†	5.22e-3(1.39e-4)†	4.53e-3(7.76e-4)†	3.75e-3(4.42e-4)†	<b>3.57e-3(8.64e-5)</b>	3.64e-3(8.46e-5)
FDA2	(20,10)	7.13e-3(3.21e-4)	5.30e-3(6.99e-5)	<b>5.00e-3(1.53e-4)</b>	5.03e-3(3.82e-4)	8.20e-3(2.10e-3)	8.00e-3(3.20e-3)
	(25,10)	7.46e-3(3.77e-4)	5.41e-3(2.32e-4)	4.87e-3(1.58e-4)	<b>4.85e-3(2.21e-4)</b>	6.41e-3(1.73e-3)	8.07e-3(1.44e-3)
	(30,10)	6.74e-3(2.18e-4)	5.41e-3(1.66e-4)	<b>4.56e-3(7.46e-5)</b>	4.78e-3(8.57e-5)	5.63e-3(9.75e-4)	7.29e-3(8.84e-4)
FDA3	(20,10)	1.07e-2(1.36e-3)†	2.49e-2(6.98e-3)†	7.04e-3(7.30e-4)	<b>5.85e-3(3.13e-4)</b>	3.32e-2(1.53e-3)	9.96e-3(1.56e-3)
	(25,10)	7.61e-3(3.82e-4)	2.77e-2(6.19e-3)†	6.06e-3(7.45e-4)	<b>5.55e-3(3.99e-4)</b>	3.07e-2(1.33e-3)	8.50e-3(1.59e-3)
	(30,10)	5.96e-3(4.09e-4)	2.92e-2(1.17e-3)†	5.51e-3(7.25e-4)	<b>5.02e-3(3.90e-4)</b>	3.24e-2(9.92e-4)†	1.12e-2(2.43e-3)
FDA4	(20,10)	1.84e-2(1.23e-3)	5.84e-2(3.68e-3)	4.15e-2(6.07e-4)†	3.95e-2(6.80e-4)†	<b>2.87e-2(1.41e-3)</b>	3.67e-2(4.50e-4)
	(25,10)	4.47e-2(5.59e-4)†	5.08e-2(2.85e-3)†	4.01e-2(6.84e-4)†	3.92e-2(9.78e-4)†	<b>2.58e-2(1.19e-3)</b>	3.64e-2(4.67e-4)
	(30,10)	4.12e-2(9.41e-4)†	4.74e-2(1.73e-3)†	4.13e-2(1.51e-3)†	3.84e-2(1.10e-3)†	<b>2.42e-2(4.78e-4)</b>	3.66e-2(2.62e-4)
dMOP1	(20,10)	1.97e-2(2.43e-3)†	9.98e-3(3.38e-3)†	5.99e-3(3.03e-3)†	7.89e-3(9.41e-3)†	4.01e-3(3.40e-4)†	<b>3.60e-3(1.97e-4)</b>
	(25,10)	1.15e-2(2.43e-3)†	6.01e-3(6.01e-4)†	4.72e-3(1.78e-3)†	<b>3.42e-3(3.04e-4)</b>	3.85e-3(3.09e-4)†	3.63e-3(9.27e-5)
	(30,10)	7.14e-3(5.55e-4)†	5.55e-3(4.60e-4)†	<b>3.11e-3(7.71e-5)</b>	<b>3.11e-3(2.93e-5)</b>	3.17e-3(1.35e-4)	3.67e-3(3.71e-5)
dMOP2	(20,10)	1.05e-2(1.06e-3)	1.02e-2(1.50e-3)	6.33e-3(6.37e-4)	5.38e-3(1.91e-3)†	4.78e-3(2.83e-4)†	<b>3.46e-3(1.86e-4)</b>
	(25,10)	7.52e-3(6.16e-4)†	7.91e-3(7.51e-4)†	3.93e-3(2.34e-4)†	4.50e-3(4.31e-4)†	4.26e-3(1.94e-4)†	<b>3.52e-3(1.43e-4)</b>
	(30,10)	6.53e-3(7.55e-4)†	7.90e-3(1.48e-3)†	4.10e-3(4.39e-4)†	3.62e-3(1.87e-4)†	3.65e-3(1.20e-4)†	<b>3.52e-3(7.17e-5)</b>
dMOP3	(20,10)	1.01e-2(3.53e-4)†	5.52e-3(2.24e-4)†	6.37e-3(2.10e-3)†	4.78e-3(4.16e-4)†	4.21e-3(1.27e-4)†	<b>3.63e-3(5.08e-5)</b>
	(25,10)	8.31e-3(1.32e-3)†	5.54e-3(4.22e-4)†	4.99e-3(1.41e-3)†	3.76e-3(3.13e-4)†	3.94e-3(7.22e-5)†	<b>3.67e-3(7.85e-5)</b>
	(30,10)	6.45e-3(3.89e-4)†	5.26e-3(9.22e-5)†	3.86e-3(8.15e-4)†	3.74e-3(3.75e-4)†	<b>3.57e-3(8.64e-5)</b>	3.71e-3(9.99e-5)
JY1	(20,10)	1.36e-2(1.06e-3)	3.48e-1(2.51e-2)†	1.08e-2(1.31e-3)†	8.09e-3(1.24e-3)†	1.01e-2(2.60e-4)†	<b>5.07e-3(2.80e-4)</b>
	(25,10)	9.87e-3(8.88e-4)†	3.46e-1(3.30e-2)†	6.80e-3(2.68e-3)†	7.33e-3(4.20e-4)†	9.06e-3(1.48e-4)†	<b>5.00e-3(8.74e-5)</b>
	(30,10)	9.41e-3(2.28e-4)†	3.78e-1(2.79e-2)†	5.93e-3(1.52e-3)†	6.11e-3(3.35e-4)†	8.65e-3(2.81e-4)†	<b>4.91e-3(1.88e-4)</b>
JY2	(20,10)	1.36e-2(1.60e-3)†	1.83e-2(2.40e-3)†	1.01e-2(4.57e-3)†	6.19e-3(8.01e-4)†	6.99e-3(1.70e-4)†	<b>3.86e-3(9.94e-5)</b>
	(25,10)	9.58e-3(7.72e-4)†	1.75e-2(3.75e-3)†	6.55e-3(2.13e-3)†	6.06e-3(7.57e-4)†	6.13e-3(4.43e-4)†	<b>3.89e-3(2.08e-4)</b>
	(30,10)	8.78e-3(6.01e-4)†	2.03e-2(3.87e-3)†	5.54e-3(1.72e-3)†	4.92e-3(6.00e-4)†	5.54e-3(4.67e-4)†	<b>3.89e-3(7.04e-5)</b>
JY3	(20,10)	1.81e-2(1.13e-3)	4.76e-2(1.94e-2)†	1.69e-2(2.70e-3)†	1.52e-2(3.25e-3)†	1.15e-2(1.35e-3)†	<b>7.16e-3(6.18e-4)</b>
	(25,10)	1.49e-2(1.51e-3)†	4.65e-2(8.07e-3)†	1.27e-2(1.75e-3)†	1.51e-2(3.01e-3)†	1.17e-2(2.03e-3)†	<b>6.54e-3(2.64e-4)</b>
	(30,10)	1.42e-2(1.29e-3)†	3.07e-2(1.35e-2)†	9.47e-3(9.22e-4)†	1.18e-2(2.45e-3)†	1.02e-2(1.00e-3)†	<b>6.52e-3(3.77e-4)</b>
JY4	(20,10)	1.02e-2(4.92e-4)†	2.85e-1(3.55e-2)†	1.07e-2(1.36e-3)†	<b>6.48e-3(5.30e-4)</b>	1.84e-2(4.34e-4)†	9.14e-3(6.72e-4)
	(25,10)	6.93e-3(3.90e-4)	2.83e-1(1.32e-2)†	8.26e-3(1.65e-3)	<b>6.26e-3(6.08e-4)</b>	1.73e-2(1.33e-3)†	8.98e-3(2.04e-4)
	(30,10)	<b>5.03e-3(3.02e-4)</b>	2.93e-1(1.17e-2)†	6.26e-3(6.33e-4)	5.91e-3(9.09e-4)	1.66e-2(7.13e-4)†	8.76e-3(6.52e-4)
JY5	(20,10)	1.15e-2(7.96e-4)†	6.83e-2(1.03e-2)†	4.44e-3(5.42e-4)†	<b>3.63e-3(9.34e-5)</b>	3.71e-3(3.56e-4)	3.85e-3(8.84e-5)
	(25,10)	9.60e-3(8.50e-4)†	7.37e-2(1.01e-2)†	3.59e-3(2.33e-4)	3.74e-3(2.95e-4)	<b>3.55e-3(4.44e-4)</b>	3.83e-3(6.39e-5)
	(30,10)	7.51e-3(3.57e-4)†	9.40e-2(1.32e-2)†	3.94e-3(3.89e-4)†	<b>3.57e-3(2.30e-4)</b>	3.68e-3(6.82e-4)	3.81e-3(4.22e-5)
JY6	(20,10)	1.35e-1(1.64e-2)†	1.72e-1(3.72e-3)†	2.51e-1(2.33e-2)†	2.33e-1(1.86e-2)†	7.12e-2(8.16e-3)†	<b>2.49e-2(1.96e-3)</b>
	(25,10)	8.15e-2(5.21e-3)†	1.25e-1(8.23e-3)†	2.29e-1(1.76e-2)†	1.90e-1(1.61e-2)†	4.90e-2(4.73e-3)†	<b>2.31e-2(2.58e-3)</b>
	(30,10)	6.12e-2(4.80e-3)†	1.04e-1(6.05e-3)†	2.00e-1(1.64e-2)†	1.70e-1(1.47e-2)†	3.40e-2(3.09e-3)†	<b>2.17e-2(2.84e-3)</b>
JY7	(20,10)	1.78e-1(1.35e-2)†	2.85e-1(3.36e-2)†	1.34e+0(6.00e-1)†	9.76e-1(1.50e-1)†	1.05e-1(2.19e-2)†	<b>4.04e-2(1.08e-2)</b>
	(25,10)	1.09e-1(2.32e-2)†	2.09e-1(2.69e-2)†	1.19e+0(5.04e-1)†	8.18e-1(2.33e-1)†	6.80e-2(1.20e-2)†	<b>3.81e-2(9.16e-3)</b>
	(30,10)	7.94e-2(1.51e-2)†	1.57e-1(1.82e-2)†	5.33e-1(6.29e-1)†	7.48e-1(1.55e-1)†	5.87e-2(2.10e-2)†	<b>3.13e-2(1.09e-2)</b>
JY8	(20,10)	1.08e-2(4.30e-4)†	1.58e-2(7.66e-3)†	4.89e-3(1.57e-4)	<b>4.53e-3(2.95e-4)</b>	7.44e-3(6.19e-4)†	5.45e-3(3.99e-4)
	(25,10)	9.35e-3(4.02e-4)†	3.30e-2(1.67e-2)†	<b>4.58e-3(1.84e-4)</b>	4.61e-3(2.65e-4)	7.27e-3(3.17e-4)†	5.23e-3(2.63e-4)
	(30,10)	1.01e-2(3.70e-4)†	2.22e-2(1.33e-2)†	4.63e-3(2.27e-4)	<b>4.46e-3(1.60e-4)</b>	6.58e-3(3.52e-4)†	5.18e-3(3.93e-4)
JY9	(20,10)	3.78e-2(3.65e-3)†	1.39e+0(5.79e-2)†	1.13e-2(1.41e-3)†	2.43e-2(8.68e-3)†	1.08e-2(2.01e-3)†	<b>6.81e-3(2.13e-4)</b>
	(25,10)	1.66e-2(1.13e-3)†	1.47e+0(1.22e-1)†	1.04e-2(3.09e-3)†	1.72e-2(2.80e-3)†	7.46e-3(7.32e-4)†	<b>7.40e-3(1.89e-3)</b>
	(30,10)	1.16e-2(1.79e-3)†	1.47e+0(1.78e-1)†	6.89e-3(1.55e-3)†	1.33e-2(1.32e-3)†	<b>5.10e-3(4.20e-4)</b>	6.58e-3(1.68e-4)

‡ and † indicate DVA performs significantly better than and equivalently to the corresponding algorithm, respectively.

### 5.3. Comparison of the distribution of the final obtained population

In order to analyze the performance of the final population, we randomly selected four test problems from the FDA, dMOP and JY test suites, including FDA1, dMOP2, JY2 and JY6. The final population gained by six algorithms at different stages are shown in figures 7, 8, 9 and 10.

From Figure.7, it is clear that all of the algorithms on FDA1 have better distribution and convergence except for RIS and PPS. This indicates that the RIS strategy alone cannot solve dMOP problems very well. As for PPS, due to the lack of historical information in the early stages, the algorithm cannot converge, but with an increasing amount of information, the distribution and

convergence of the algorithm improves. Unlike FDA1, the *POF* of dMOP2 changes when the environment changes. We show the *POF* and population at different times in Fig.8. We can clearly see that RIS and DNSGA-II cannot solve this kind of problem very well. Similar to FDA1, the performance of PPS at early stages was not good. The distribution of DMS is better than SGEA, but the convergence of SGEA is better than DMS. DVA was superior to SGEA and DMS in both distribution and convergence except when time  $t = 10$ . The reason is that when  $t = 10$ , the environment changes dramatically, while DVA does not provide a better initial population, which results in a decrease in the convergence rate of the algorithm. On the JY test suites, as can be seen from Fig.9 and Fig.10, DMS and SGEA performed well, but slightly worse than DVA.

## 6. Discussion

In this section, we mainly discuss the impact of different parts of the algorithm on the whole. The DVA algorithm consists of a response strategy and DEE. The response strategy consists of two main components, one is the prediction part, and the other is the adaptive selection part of the two populations based on the previous prediction part. To deeply examine the role that each component plays in dynamic optimization, we created two variants of the original DVA. The first variant(S1) does not use the adaptive selection which means that the initial population consists entirely of individuals produced by the prediction component. The second variant(S2) contains a complete response strategy, but the static part of the algorithm uses NSGA-II instead of DEE. On the one hand, this is to verify the impact of DEE on the whole; on the other hand, it is also to verify whether our proposed strategy can effectively solve DMOPs when combined with other algorithms. These two variants are compared with the original DVA on four problems with settings of  $(\tau, n_t)=(20,10)$ . Table.4 presents the average and standard deviations of the metrics obtained by DVA and its variants. On FDA1, JY1 and JY9, it can be clearly seen that s1 is not as good as the original DVA in convergence and distribution, but it

is obviously better than DVA-S2. Combined with Tables 1, 2 and 3 above, DVA containing only the prediction part is still significantly better than other algorithms, which indicates that the DVA prediction part can effectively handle DMOPs. On dMOP1, DVA-S1 has SP and IGD values similar to the original DVA, but the GD of DVA-S1 is significantly smaller than the original DVA. In general, on dDMOP1, DVA-S1 were better than the original DVA and DVA-S2 because the problem of the first two random initializations is too dense on the dMOP1 problem. In order to balance the distribution as much as possible, the original DVA sacrifices the solution with good convergence, and those solutions with relatively good distribution are selected for the initial population. Thus, DVA-S1 is obviously better than the original DVA in convergence. We next analyze the second variant of DVA. Although DVA-S2 is not as good as DVA-S1 and the original DVA in terms of convergence and distribution, DVA-S2 was obviously superior to DNSGA-II under the same static algorithm on dMOP1, JY1 and JY9. This means that our proposed response strategy combined with other related static algorithms can also effectively solve DMOPs.

Table 4: Mean and SD of GD, IGD and SP indicators of DVA variants

Prob	Indicator	DVA-S1	DVA-S2	DVA
FDA1	GD	3.00e-3(9.31e-4)	5.10e-3(2.87e-4)	<b>2.12e-3(1.72e-4)</b>
	IGD	5.18e-3(7.59e-4)	8.22e-3(5.03e-4)	<b>4.48e-3(1.43e-4)</b>
	SP	3.74e-3(1.86e-4)	4.93e-3(1.03e-3)	<b>3.66e-3(8.95e-5)</b>
dMOP1	GD	<b>5.47e-3(7.54e-3)</b>	1.28e-2(5.48e-3)	9.23e-3(1.11e-2)
	IGD	<b>1.48e-2(9.78e-3)</b>	1.89e-2(7.78e-3)	1.52e-2(1.55e-2)
	SP	<b>3.45e-3(7.67e-5)</b>	4.04e-3(3.52e-4)	3.55e-3(1.25e-4)
JY1	GD	2.88e-3(3.32e-4)	4.72e-3(4.55e-4)	<b>2.48e-3(1.04e-3)</b>
	IGD	6.42e-3(2.93e-4)	1.03e-2(6.74e-4)	<b>6.08e-3(1.07e-3)</b>
	SP	5.00e-3(1.49e-4)	6.35e-3(4.83e-4)	<b>4.86e-3(3.08e-4)</b>
JY9	GD	1.28e-2(2.13e-3)	1.51e-2(3.01e-3)	<b>1.00e-2(2.42e-3)</b>
	IGD	1.48e-2(1.82e-3)	2.02e-2(2.10e-3)	<b>1.45e-2(2.18e-3)</b>
	SP	6.93e-3(4.76e-4)	9.16e-3(1.86e-3)	<b>6.68e-3(3.50e-4)</b>

In addition, in order to verify the validity of DVA under the same benchmarks, we compared DVA with a dynamic multi-objective evolutionary algorithm based on intensity of environmental change(IEC) [52]. In IEC, the algorithm

effectively tracks the POF according to the intensity of environmental change,  
 595 and in this paper, JY is also used as the standard test problem set. The specific  
 experimental results are shown in Table 5, and we can intuitively see that in  
 JY1, JY5, JY6, JY7 and JY8, the performance of DVA is significantly better than  
 IEC.

Table 5: Mean and SD of IGD of IEC and DVA

Prob	$(n_t, \tau_t)$	IEC [52]	DVA
JY1	(10,30)	6.11e-3(4.03e-5)	<b>5.94e-3(4.33e-4)</b>
JY2	(10,30)	<b>4.98e-2(3.92e-5)</b>	5.04e-2(2.76e-4)
JY3	(10,30)	<b>3.10e-1(1.64e-3)</b>	3.12e-1(1.76e-3)
JY4	(10,30)	<b>1.97e-2(6.75e-5)</b>	2.08e-2(3.61e-4)
JY5	(10,30)	6.52e-3(6.41e-5)	<b>4.22e-3(5.34e-5)</b>
JY6	(10,30)	4.03e-1(6.38e-2)	<b>3.25e-1(5.31e-2)</b>
JY7	(10,30)	2.02e+0(4.86e-1)	<b>5.30e-1(4.06e-1)</b>
JY8	(10,30)	1.51e-2(2.67e-4)	<b>7.52e-3(2.84e-4)</b>
JY9	(10,30)	<b>6.66e-3(1.19e-4)</b>	1.36e-2(1.28e-3)

## 7. Conclusions and future work

600 In order to quickly respond to environmental changes and make the popu-  
 lation adapt to the new environment better, we propose DVA to solve DMOPs.  
 DVA as a response strategy guides the whole population to evolve to the nex-  
 t *POF* when environmental changes are detected. Although the prediction  
 may be inaccurate, dimension-based analysis ensures population diversity. The  
 605 idea of the DVA algorithm is to analyze the impact of each decision variable  
 on individuals in the current environment, find those dimensions that affect

the distribution, combine with historical information, adopt different prediction methods to generate solutions, and then select those solutions which have good convergence and distribution in the current environment to form the initial population. Compared with five other DMOEAs, experiments show that DVA responds more quickly to environmental changes in those problems with non-linear relationships of decision variables, and it responds better to most of the problems when environmental changes are relatively stable, especially when dealing with complex non-linear problems.

Several extensions are possible for future work:

- Although DVA has great advantages compared with other algorithms, it lacks the correlation analysis between decision variables before and after environmental changes, so it is necessary to design corresponding methods to analyze the correlation between them.
- the DMOEAs have demonstrated the ability to handle constrained DMOPs and solving constrained DMOPs is a prospective research problem. Simultaneously, new dynamic benchmarks and performance metrics are needed to evaluate the performance of the algorithm.
- DMOEAs should be more applied in practical applications in other fields, such as scheduling [10], control [11] [12] and planning [13] [53].

## Acknowledgement

This work was supported by the research projects: the National Natural Science Foundation of China under Grant Nos. 61502408, 61673331, 61772178 and 61403326, the postgraduate research and innovation Project of Hunan Province under Grant No. XDCX2019B057.



## References

- [1] M. Farina, K. Deb, P. Amato, Dynamic multiobjective optimization problems: test cases, approximations, and applications, *Evolutionary Computation*, IEEE Transactions on 8 (5) (2004) 425–442.
- 635 [2] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: Nsga-ii, *Evolutionary Computation*, IEEE Transactions on 6 (2) (2002) 182–197.
- [3] E. Zitzler, M. Laumanns, L. Thiele, Spea2: Improving the strength pareto evolutionary algorithm for multiobjective optimization, *Evolutionary*  
640 *Methods for Design, Optimization, and Control* (2002) 95–100.
- [4] S.-y. Zeng, G. Chen, L. Zheng, H. Shi, H. De Garis, L. Ding, L. Kang, A dynamic multi-objective evolutionary algorithm based on an orthogonal design, in: *Evolutionary Computation, 2006. CEC 2006. IEEE Congress on, IEEE, 2006*, pp. 573–580.
- 645 [5] A. Zhou, Y. Jin, Q. Zhang, B. Sendhoff, E. Tsang, Prediction-based population re-initialization for evolutionary dynamic multi-objective optimization, in: *Evolutionary Multi-Criterion Optimization*, Springer, 2007, pp. 832–846.
- [6] C.-a. Liu, Y. Wang, New evolutionary algorithm for dynamic multiobjective optimization problems, in: *Advances in Natural Computation*, Springer,  
650 2006, pp. 889–892.
- [7] H. Richter, Detecting change in dynamic fitness landscapes, in: *Evolutionary Computation, 2009. CEC’09. IEEE Congress on, IEEE, 2009*, pp. 1613–1620.
- 655 [8] Y. Wu, Y. Jin, X. Liu, A directed search strategy for evolutionary dynamic multiobjective optimization, *Soft Computing* (2014) 1–15.

- [9] K. Deb, S. Karthik, et al., Dynamic multi-objective optimization and decision-making using modified nsga-ii: a case study on hydro-thermal power scheduling, in: *Evolutionary Multi-Criterion Optimization*, Springer, 2007, pp. 803–817.
- [10] M. B. Abello, L. T. Bui, Z. Michalewicz, An adaptive approach for solving dynamic scheduling with time-varying number of tasks part ii, in: *Evolutionary Computation (CEC)*, 2011 IEEE Congress on, IEEE, 2011, pp. 1711–1718.
- [11] B. Andres-Toro, J. Giron-Sierra, P. Fernandez-Blanco, J. Lopez-Orozco, E. Besada-Portas, Multiobjective optimization and multivariable control of the beer fermentation process with the use of evolutionary algorithms, *Journal of Zhejiang University SCIENCE* 5 (4) (2004) 378–389.
- [12] Z. Zhang, Multiobjective optimization immune algorithm in dynamic environments and its application to greenhouse control, *Applied Soft Computing* 8 (2) (2008) 959–971.
- [13] L. T. Bui, Z. Michalewicz, An evolutionary multi-objective approach for dynamic mission planning, in: *Evolutionary Computation (CEC)*, 2010 IEEE Congress on, IEEE, 2010, pp. 1–8.
- [14] A. R. da Cruz, R. T. Cardoso, R. H. Takahashi, Multiobjective dynamic optimization of vaccination campaigns using convex quadratic approximation local search, in: *Evolutionary Multi-Criterion Optimization*, Springer, 2011, pp. 404–417.
- [15] P. Di Barba, Dynamic multiobjective optimization: A way to the shape design with transient magnetic fields, *IEEE transactions on magnetics* 44 (6) (2008) 962–965.
- [16] Y. Jin, J. Branke, Evolutionary optimization in uncertain environments—a survey, *Evolutionary Computation, IEEE Transactions on* 9 (3) (2005) 303–317.

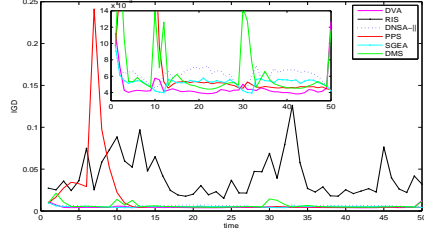
- 685 [17] S. Jiang, S. Yang, Evolutionary dynamic multiobjective optimization: Benchmarks and algorithm comparisons, *IEEE transactions on cybernetics* 47 (1) (2017) 198–211.
- [18] S. B. Gee, K. C. Tan, C. Alippi, Solving multiobjective optimization problems in unknown dynamic environments: An inverse modeling approach, *IEEE transactions on cybernetics* 47 (12) (2017) 4223–4234.
- 690 [19] J. Zou, Q. Li, S. Yang, H. Bai, J. Zheng, A prediction strategy based on center points and knee points for evolutionary dynamic multi-objective optimization, *Applied Soft Computing* 61 (2017) 806–818.
- [20] J. Zou, Q. Li, S. Yang, J. Zheng, Z. Peng, T. Pei, A dynamic multiobjective evolutionary algorithm based on a dynamic evolutionary environment model, *Swarm and Evolutionary Computation*.
- 695 [21] A. Muruganantham, K. C. Tan, P. Vadakkepat, Evolutionary dynamic multiobjective optimization via kalman filter prediction, *IEEE Trans Cybern* 46 (12) (2015) 2862.
- 700 [22] M. Jiang, Z. Huang, L. Qiu, W. Huang, G. G. Yen, Transfer learning-based dynamic multiobjective optimization algorithms, *IEEE Transactions on Evolutionary Computation* 22 (4) (2018) 501–514.
- [23] R. Chen, K. Li, X. Yao, Dynamic multiobjectives optimization with a changing number of objectives, *IEEE Transactions on Evolutionary Computation* 22 (1) (2018) 157–171.
- 705 [24] X.-F. Liu, Y.-R. Zhou, X. Yu, Cooperative particle swarm optimization with reference-point-based prediction strategy for dynamic multiobjective optimization, *Applied Soft Computing* 87 (2020) 105988.
- [25] I. Hatzakis, D. Wallace, Dynamic multi-objective optimization with evolutionary algorithms: a forward-looking approach, in: *Proceedings of the 8th annual conference on Genetic and evolutionary computation*, ACM, 2006, pp. 1201–1208.
- 710

- [26] Y. Wang, B. Li, Investigation of memory-based multi-objective optimization evolutionary algorithm in dynamic environment, in: *Evolutionary Computation*, 2009. CEC'09. IEEE Congress on, IEEE, 2009, pp. 630–637.
- [27] Z. Peng, J. Zheng, J. Zou, M. Liu, Novel prediction and memory strategies for dynamic multiobjective optimization, *Soft Computing* 19 (9) (2015) 2633–2653.
- [28] Z. Liang, S. Zheng, Z. Zhu, S. Yang, Hybrid of memory and prediction strategies for dynamic multiobjective optimization, *Information Sciences* 485 (2019) 200–218.
- [29] W. T. Koo, C. K. Goh, K. C. Tan, A predictive gradient strategy for multi-objective evolutionary algorithms in a fast changing environment, *Memetic Computing* 2 (2) (2010) 87–110.
- [30] A. Muruganantham, Y. Zhao, S. B. Gee, Dynamic multiobjective optimization using evolutionary algorithm with kalman filter, *Procedia Computer Science* 24 (2013) 66–75.
- [31] F. Zou, G. G. Yen, L. Tang, A knee-guided prediction approach for dynamic multi-objective optimization, *Information Sciences* 509 (2020) 193–209.
- [32] Q. Zhang, S. Yang, S. Jiang, R. Wang, X. Li, Novel prediction strategies for dynamic multiobjective optimization, *IEEE Transactions on Evolutionary Computation* 24 (2) (2019) 260–274.
- [33] Q. Li, J. Zou, S. Yang, J. Zheng, G. Ruan, A predictive strategy based on special points for evolutionary dynamic multi-objective optimization, *Soft Computing* 23 (11) (2019) 3723–3739.
- [34] L. Cao, L. Xu, E. D. Goodman, H. Li, Decomposition-based evolutionary dynamic multiobjective optimization using a difference model, *Applied Soft Computing* 76 (2019) 473–490.

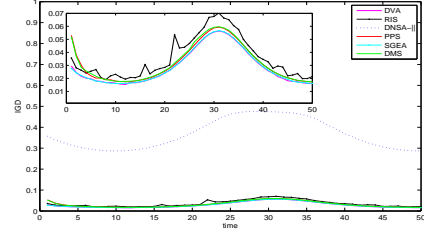
- [35] K. Kim, R. I. McKay, B.-R. Moon, Multiobjective evolutionary algorithms  
740 for dynamic social network clustering, in: Proceedings of the 12th annual conference on Genetic and evolutionary computation, ACM, 2010, pp. 1179–1186.
- [36] A. Díaz-Manríquez, G. T. Pulido, J. G. Ramírez-Torres, Handling dynamic  
745 multiobjective problems with particle swarm optimization., in: ICAART (1), 2010, pp. 337–342.
- [37] Y.-J. Zhang, S.-F. Shao, J. Niyongabo, Cloud hyper mutation particle swarm optimization algorithm based on cloud model, Pattern Recognition and Artificial Intelligence 24 (1) (2011) 90–94.
- [38] V. S. Aragón, S. C. Esquivel, C. Coello Coello, Evolutionary multiobjective  
750 optimization in non-stationary environments, Journal of Computer Science & Technology 5.
- [39] C.-K. Goh, K. C. Tan, A competitive-cooperative coevolutionary paradigm for dynamic multiobjective optimization, Evolutionary Computation, IEEE Transactions on 13 (1) (2009) 103–127.
- [40] K. R. Harrison, B. M. Ombuki-Berman, A. P. Engelbrecht, Dynamic multi-  
755 objective optimization using charged vector evaluated particle swarm optimization, in: IEEE Congress on Evolutionary Computation, 2014.
- [41] M. Helbig, A. P. Engelbrecht, Archive management for dynamic multi-objective optimisation problems using vector evaluated particle swarm optimisation, in: Evolutionary Computation (CEC), 2011 IEEE Congress on,  
760 IEEE, 2011, pp. 2047–2054.
- [42] I. Hatzakis, D. Wallace, Topology of anticipatory populations for evolutionary dynamic multi-objective optimization, in: 11th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, 2006, p. 7071.

- 765 [43] A. Zhou, Y. Jin, Q. Zhang, A population prediction strategy for evolutionary dynamic multiobjective optimization, *Cybernetics, IEEE Transactions on* 44 (1) (2014) 40–53.
- [44] G. Ruan, G. Yu, J. Zheng, J. Zou, S. Yang, The effect of diversity maintenance on prediction in dynamic multi-objective optimization, *Applied Soft Computing* 58 (2017) 631–647.
- 770 [45] S. Jiang, S. Yang, A steady-state and generational evolutionary algorithm for dynamic multiobjective optimization, *IEEE Transactions on Evolutionary Computation* PP (99) (2017) 1–1.
- [46] M. D. McKay, R. J. Beckman, W. J. Conover, A comparison of three methods for selecting values of input variables in the analysis of output from a computer code, *Technometrics* 42 (1) (2000) 55–61.
- 775 [47] A. Zhou, Estimation of distribution algorithms for continuous multiobjective optimization, Ph.D. thesis, University of Essex (2009).
- [48] L. Cao, L. Xu, E. D. Goodman, H. Li, Decomposition-based evolutionary dynamic multiobjective optimization using a difference model, *Applied Soft Computing* 76 473–490.
- 780 [49] Z. Liang, S. Zheng, Z. Zhu, S. Yang, Hybrid of memory and prediction strategies for dynamic multiobjective optimization, *Information Sciences*.
- [50] J. R. Schott, Fault tolerant design using single and multicriteria genetic algorithm optimization., Tech. rep., DTIC Document (1995).
- 785 [51] F. Wilcoxon, Individual comparisons by ranking methods, *Biometrics bulletin* 1 (6) (1945) 80–83.
- [52] Y. Hu, J. Zheng, J. Zou, S. Yang, J. Ou, R. Wang, A dynamic multi-objective evolutionary algorithm based on intensity of environmental change, *Information Sciences*.
- 790

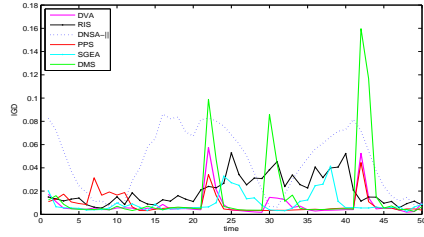
- [53] P. P. Wu, D. Campbell, T. Merz, Multi-objective four-dimensional vehicle motion planning in large dynamic environments, *Systems, Man, and Cybernetics, Part B: Cybernetics*, IEEE Transactions on 41 (3) (2011) 621–634.



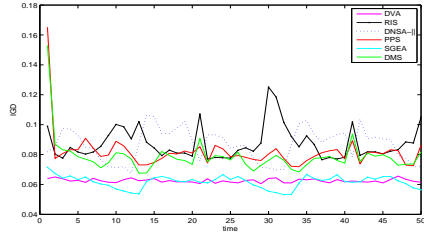
(a) FDA1



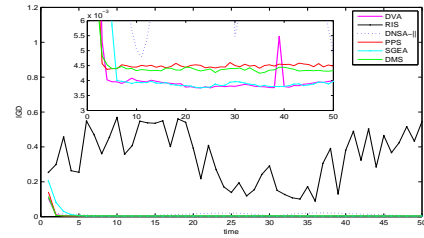
(b) FDA2



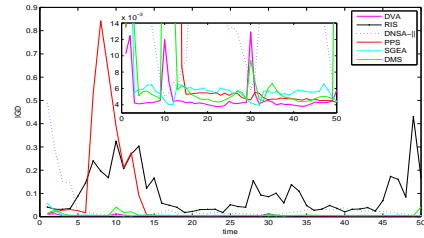
(c) FDA3



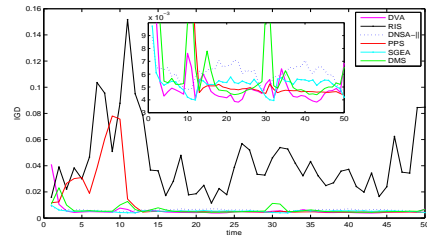
(d) FDA4



(e) dMOP1



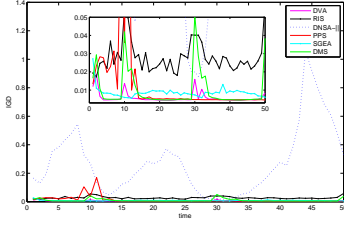
(f) dMOP2



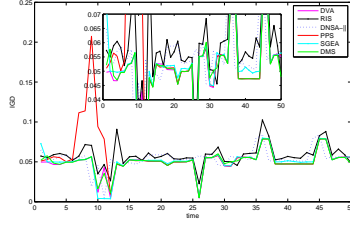
(g) dMOP3

Figure 5: Evolution curves of average IGD values for seven problems with  $n_T = 10$  and  $\tau_T = 30$ .

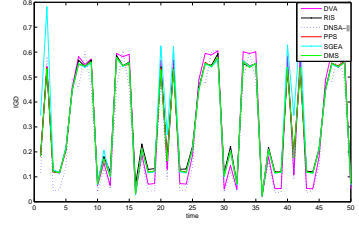




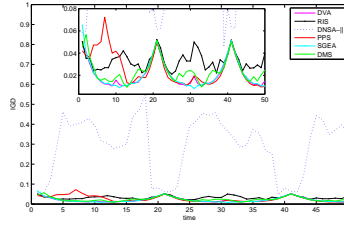
(a) JY1



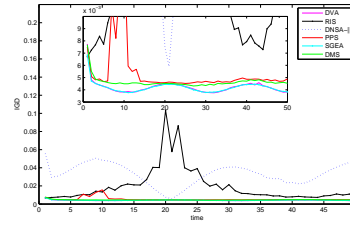
(b) JY2



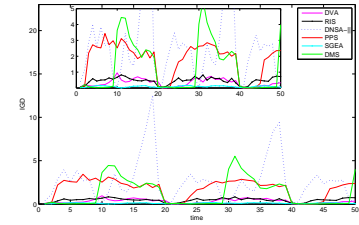
(c) JY3



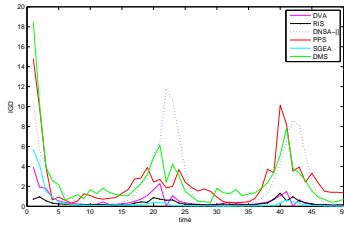
(d) JY4



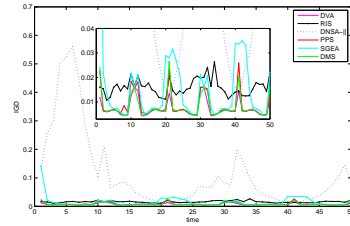
(e) JY5



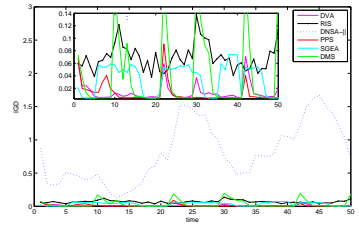
(f) JY6



(g) JY7

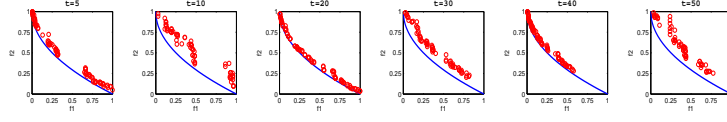


(h) JY8

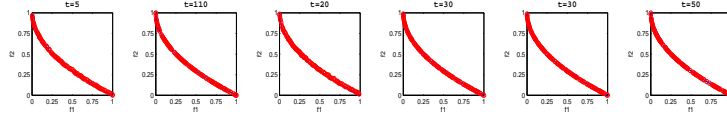


(i) JY9

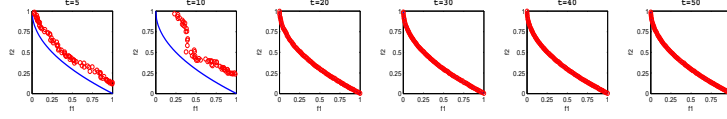
Figure 6: Evolution curves of average IGD values for nine problems with  $n_T=10$  and  $\tau_T=30$ .



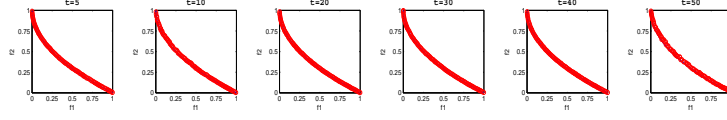
(a) RIS



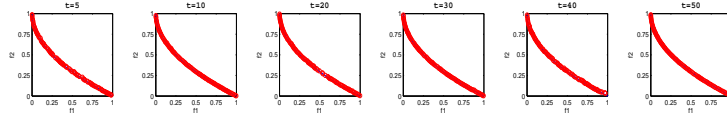
(b) DNSGA-II



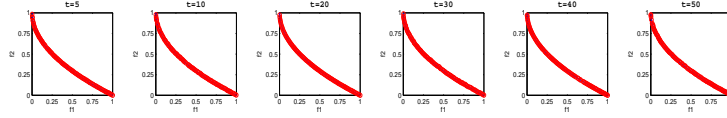
(c) PPS



(d) DMS

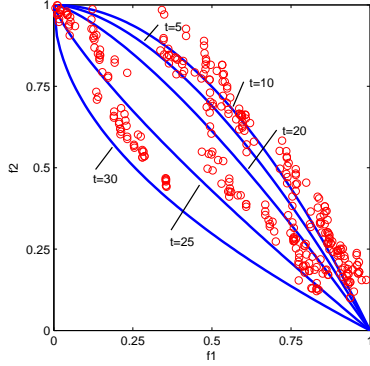


(e) SGEA

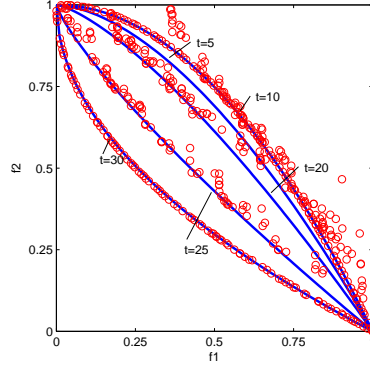


(f) DVA

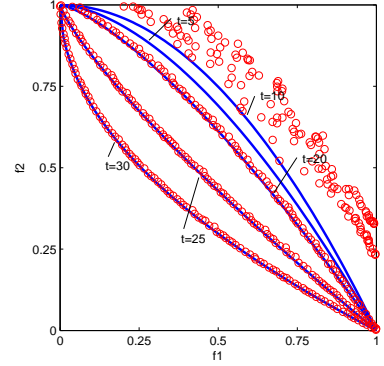
Figure 7: Solution sets obtained by six algorithms at six different time steps on FDA1.



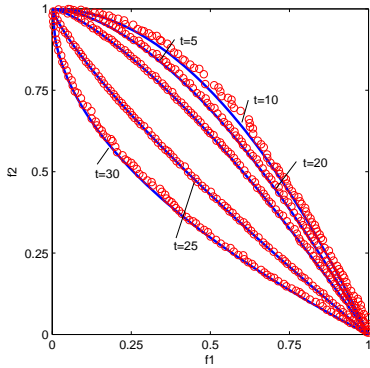
(a) RIS



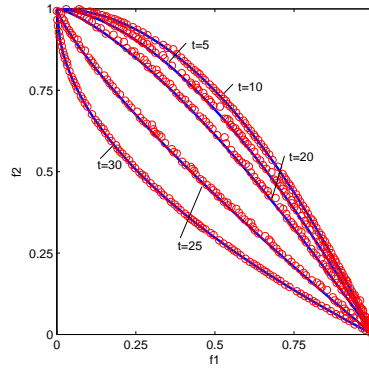
(b) DNSGA-II



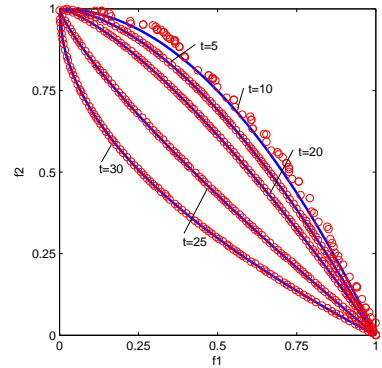
(c) PPS



(d) DMS

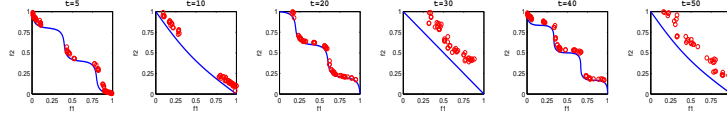


(e) SGEA

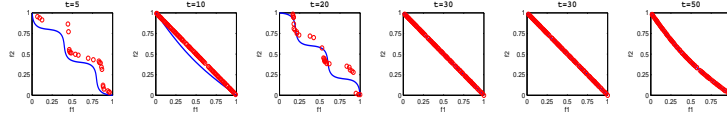


(f) DVA

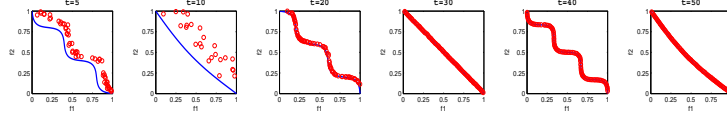
Figure 8: Solution sets obtained by six algorithms at six different time steps on dMOP2.



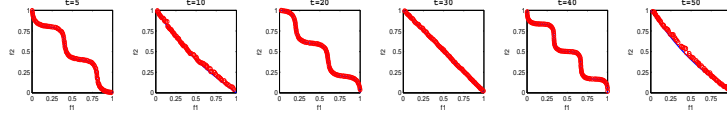
(a) RIS



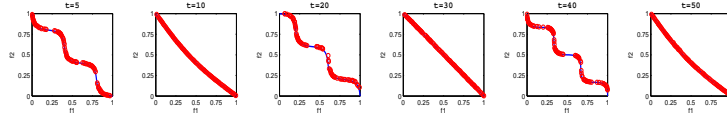
(b) DNSGA-II



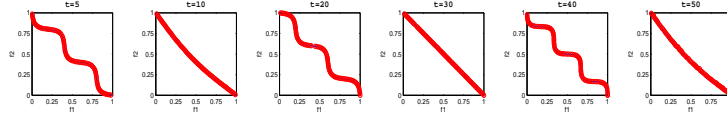
(c) PPS



(d) DMS

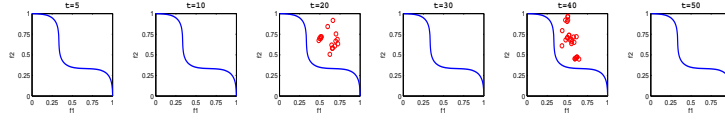


(e) SGEA

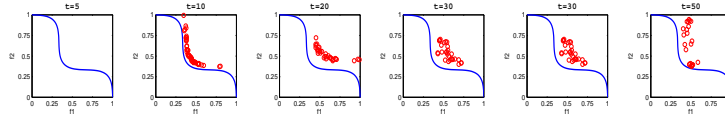


(f) DVA

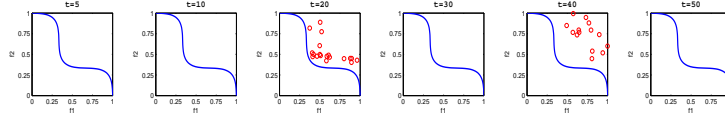
Figure 9: Solution sets obtained by seven algorithms at six different time steps on JY2.



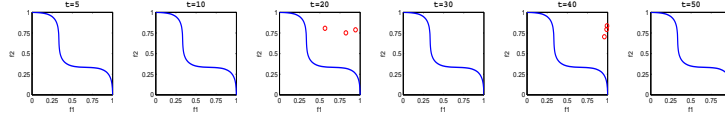
(a) RIS



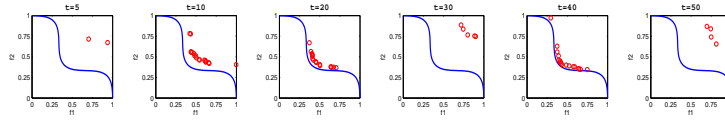
(b) DNSGA-II



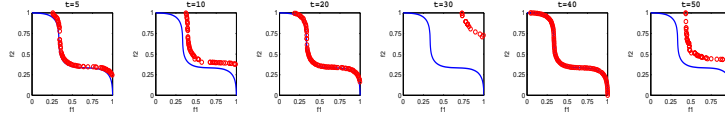
(c) PPS



(d) DMS



(e) SGEA



(f) DVA

Figure 10: Solution sets obtained by seven algorithms at six different time steps on JY6.